




Introduction	1
User Interface of Insights Hub MindConnect Nano Plugin	2
User Roles and Scopes	3
Onboarding and Configuring MindConnect Nano	4
Configuring protocols	5
Managing MindConnect Nano	6
Appendix	7
Glossary	8

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

 DANGER
indicates that death or severe personal injury will result if proper precautions are not taken.
 WARNING
indicates that death or severe personal injury may result if proper precautions are not taken.
 CAUTION
indicates that minor personal injury can result if proper precautions are not taken.
NOTICE
indicates that property damage can result if proper precautions are not taken.


If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

 WARNING
Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Table of contents

1. Introduction.....	6
1.1. Overview of MindConnect Nano.....	6
2. User Interface of Insights Hub MindConnect Nano Plugin.....	9
2.1. Insights Hub User interface of MindConnect Nano Plugin.....	9
3. User Roles and Scopes.....	12
3.1. User Roles and Scopes for MindConnect Nano.....	12
4. Onboarding and Configuring MindConnect Nano.....	13
4.1. Overview of onboarding MindConnect Nano.....	13
4.2. Logging in to Insights Hub.....	14
4.3. Creating an asset for MindConnect Nano.....	14
4.4. Enable connection to MindConnect Nano.....	16
4.5. Configuring network settings.....	17
4.6. Configuring diagnostic settings.....	19
4.7. Configuring Buffer settings.....	22
4.8. Transferring configuration to MindConnect Nano.....	24
4.9. Configuring data in Asset Manager for MindConnect Nano.....	26
4.10. Creating an aspect and variables for MindConnect Nano.....	26
4.11. Creating a type in Asset Manager for MindConnect Nano.....	26
4.12. Adding a datasource for MindConnect Nano.....	30
4.13. Adding datapoints for MindConnect Nano.....	31
4.14. Mapping an aspect to a datasource for MindConnect Nano.....	33
4.15. Browsing datapoint of a datasource for MindConnect Nano.....	36
4.16. Importing datasource configurations for MindConnect Nano.....	38
4.17. Configuring MindConnect Nano using API.....	42
5. Configuring protocols.....	46
5.1. Overview of supported protocols for MindConnect Nano.....	46

5.2. Overview of S7 protocol for MindConnect Nano.....	46
5.3. Overview of OPC UA protocol for MindConnect Nano.....	53
5.4. Overview of Modbus TCP protocol for MindConnect Nano.....	63
5.5. Overview of Modbus RTU protocol for MindConnect Nano.....	68
5.6. Overview of SYSTEM protocol for MindConnect Nano.....	75
5.7. Overview of Rockwell protocol for MindConnect Nano.....	88
6. Managing MindConnect Nano.....	91
6.1. Manual firmware update of MindConnect Nano.....	91
6.2. Firmware update in Asset Manager for MindConnect Nano.....	92
6.3. Automatic firmware update of MindConnect Nano.....	94
6.4. Remote restart for MindConnect Nano.....	95
6.5. Send commands for MindConnect Nano.....	95
6.6. USB commands.....	99
6.7. Result of CheckCloudConnection file.....	105
6.8. Troubleshooting.....	106
7. Appendix.....	118
7.1. Using quality code.....	118
7.2. Quality codes OPC UA and S7 protocol.....	119
7.3. Quality codes Modbus protocol.....	121
7.4. Quality codes SYSTEM protocol.....	122
7.5. Quality codes Rockwell protocol.....	122
7.6. Offline buffer.....	124
7.7. Read performance MindConnect Elements.....	124
7.8. Firewall Settings.....	126
7.9. Return of defective hardware.....	127
7.10. List of abbreviations.....	128
7.11. MindConnect EthernetIP - CIP Communication path (route param....	128

7.12. Case 1 – Accessing data inside an Allen- Bradley PLC.....	129
7.13. Case 2 – Accessing the Allen- Bradley PLC via integrated Ethernet ...	130
7.14. Case 3 – Accessing the Allen- Bradley PLC via the ControlNet interf...	130
7.15. MindConnect General Security Principles.....	131
7.16. Default values for datasource Import and Export.....	133
8. Glossary.....	135
8.1. Glossary.....	135

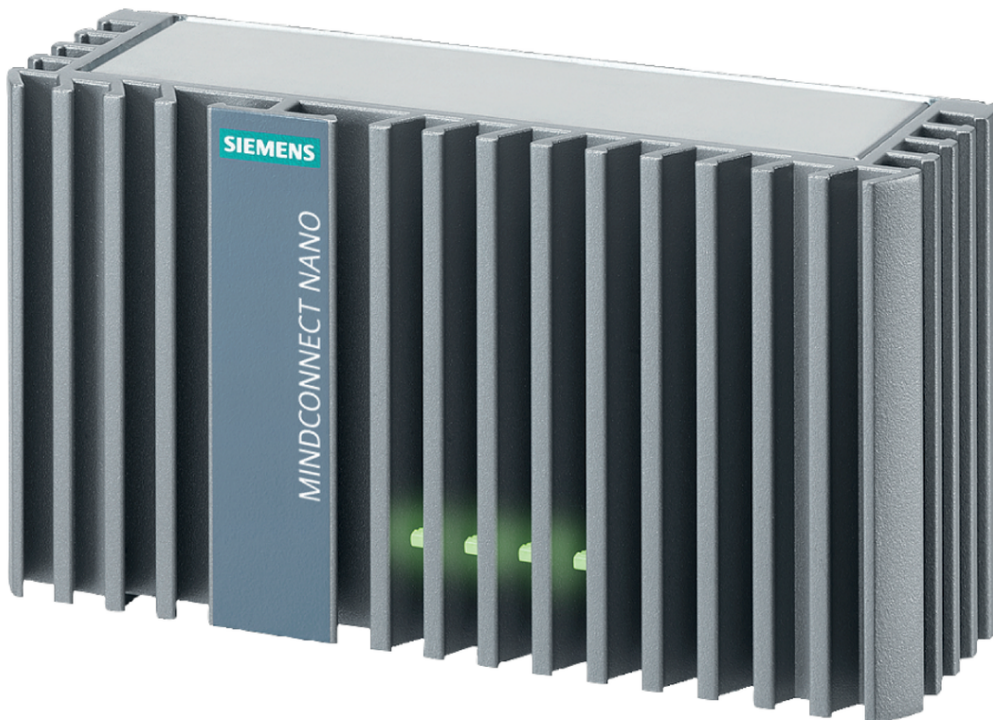
Introduction

1.1 Overview of MindConnect Nano

MindConnect Nano is an embedded industrial PC to connect industrial data sources with Industrial IoT. It allows you to connect with Industrial IoT and collect data from the field and then transfer it with encryption to Industrial IoT.







MindConnect Nano device supports the transmission of data encrypted through a secured internet connection with Industrial IoT. MindConnect Nano allows fast and easy IoT connectivity of machines and systems.

The following graphic shows the MindConnect Nano:












General Features Overview

General Features	MindConnect Nano
Basics	
Onboarding via USB Stick	✓

General Features	MindConnect Nano
Onboarding via WebServer	
Diagnostics via USB-Stick	
Diagnostics via Webserver	
Local Firmware Update (Webserver)	
Local Firmware Update (USB)	
Remote Firmware Update	
Remote Restart	
Additional Options	
Datapoint Browsing (see also data source details)	
File Transfer (File Storage or Integrated Data Lake)	
Remote Service gateway (MRS)	
Network routing	
Performance	n
Read performance (datapoints/sec)	250
Maximum number of connections (data sources)	30
Maximum number of datapoints	1050

Protocols Feature Overview

Protocols	Minimum read cycle	MindConnect Nano	Bi-Directional communication	Browsing
Basics				
S7	1 sec			
OPC UA	1 sec			
Modbus TCP IP	1 sec			

Protocols	Minimum read cycle	MindConnect Nano	Bi-Directional communication	Browsing
Modbus RTU	1 sec	✓	✗	✗
Rockwell	1 sec	✓	✗	✗
Simatic I/O Shield	1 sec	✗	✗	●
System	1 sec	✓	✗	●

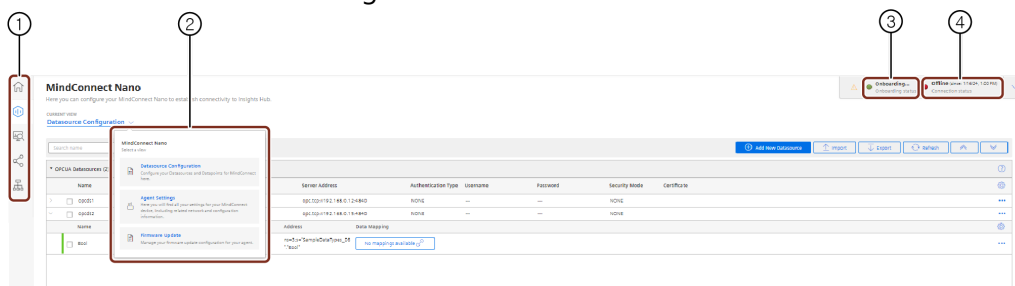


- ✓ = supported
- ✗ = not supported
- ● = in development

User Interface of Insights Hub MindConnect Nano² Plugin

2.1 Insights Hub User interface of MindConnect Nano Plugin

Manage the MindConnect Nano Plugin UI from the asset manger. The below image displays the UI of MindConnect Nano Plugin.



① Navigation menu - On the left side of the UI, menu options available to navigate between the different tabs are mentioned below.

- Home
- Assets
- Library
- Sharings
- Connectivity

② Select view for MindConnect Nano

The new UI is re-organized for all functionalities in three views i.e. "Datasource Configuration", "Agent Settings" and "Firmware Update" view. Based on the functionality, user can select the following view.

Datasource Configuration view

Datasource Configuration view enables the user to configure the datasources, datapoints and also to map the datapoints. The user can use the "Refresh" option to discard the changes made.

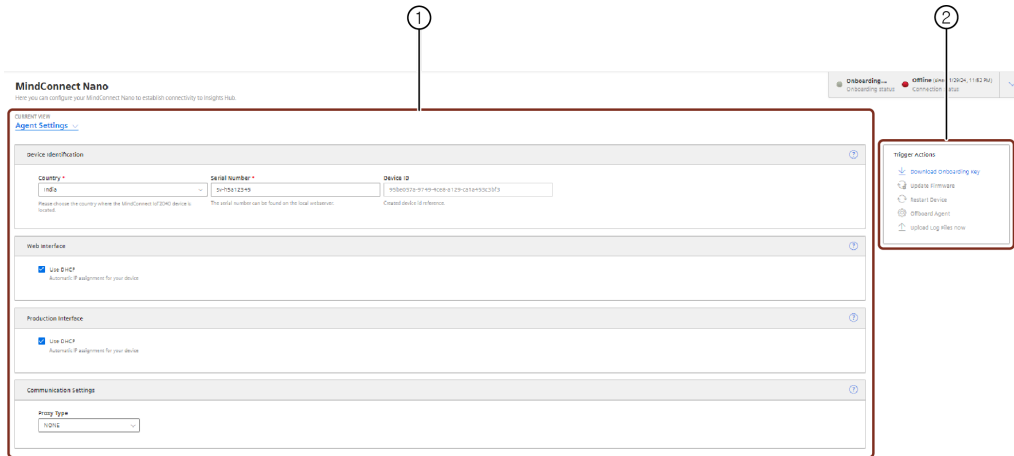


Agent Settings view

In the "Agent Settings" view, user can find all the settings for MindConnect device including the related network and configuration information which enables the user to configure the agent.

2.1 Insights Hub User interface of MindConnect Nano Plugin

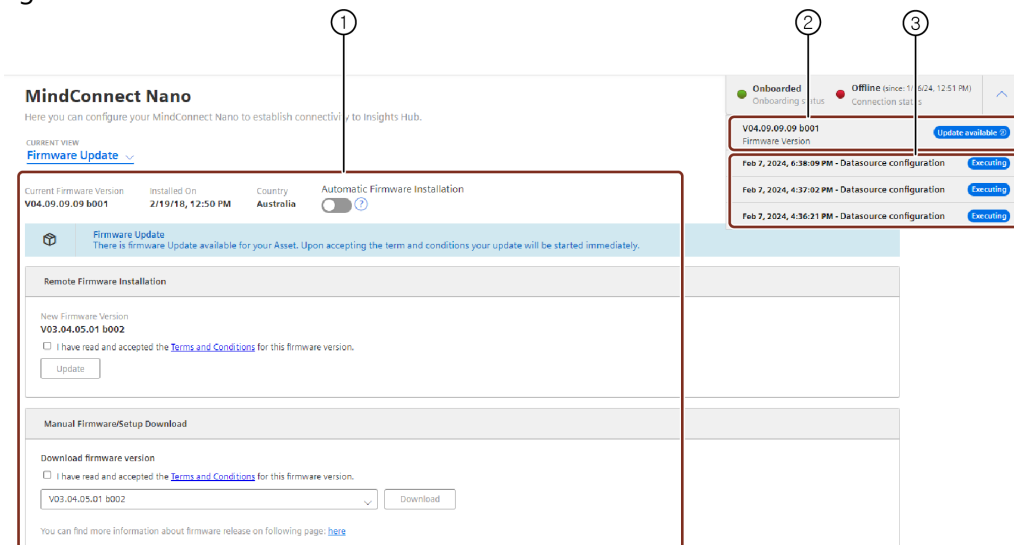
Agent settings view is shown in the below image.



- ① Agent Settings - Agents are configured or modified in this window.
- ② Trigger Actions - Following are the options available in Trigger actions window:
 - Download Onboarding Key: Downloads the Onboarding key to the local folder.
 - Update Firmware: Updates to the latest firmware.
 - Restart Device: Restarts the device.
 - Offboard Agent: Offboards the Agent from the device.
 - Upload Log Files now: Uploads all the generated logs to Insights Hub.

Firmware Update view

In the "Firmware Update" view, user can manage and update firmware configuration for the agent.



- ① Latest Firmware update is displayed
- ② Current firmware version
- ③ Displays the last three jobs status in the MindConnect Nano UI.

MindConnect Nano Plugin new UI changes

The following are the main changes made in the new UI of MindConnect Nano Plugin,

The configuration is downloaded direct to the device once saved(Apply changes is now not available).

All configuration changes are done within a single grid. In order to change any value the user can click the field and edit the changes.

On the top right, the "Jobs" icon is displayed which indicates the status. For example, a configuration is successfully deployed to the device.

The configuration grid can be changed using the gear icon.

Datasources is now mapped directly, click the mapping field and select the asset to connect. For more information, refer [Datamapping](#).

The import of datapoints (including Browsing) is now integrated in a wizard style workflow. For more information, refer to [Importing datasource](#)

Grouping of datasources is available based on protocol.

Delete all datasources functionality is discontinued. Multi-select and delete operation on datasources is now supported.

Log column is discontinued in New UI.

User Roles and Scopes

3


3.1 User Roles and Scopes for MindConnect Nano

There are multiple roles that are available for MindConnect device controlling. A device can be configured and can send the command to the device by an "Admin" user. If you want to ensure that a user cannot configure but can send a command, then you can use "DeviceCommander" role. If you want to ensure that a user can configure but cannot send a command, then you can use "DeviceConfigurator" role.

The following table shows the list of MindConnect device roles and scopes:

User role	Functionality or Scope
StandardUser (uipluginassetmanagemcnano:user)	This role grants the permission to visualize the MindConnect device plugin, but cannot change the device configuration and cannot send commands to the device.
DeviceConfigurator (uipluginassetmanagemcnano:deviceconfigurator)	This role grants the permission to visualize the MindConnect device plugin and update the device configuration, but cannot send commands to the device.
DeviceCommander (uipluginassetmanagemcnano:devicecommander)	This role grants the permission to visualize the MindConnect device plugin and can send the commands to the device, but cannot update the device configuration.
Admin (uipluginassetmanagemcnano:admin)	This role grants the permission to configure and command the MindConnect device.

By default, these roles are not enabled on the environment. To enable these roles you need to navigate to the "Settings" app.

In "Settings" app, click on "Roles" tab and click  to enable the new roles. For more information about roles, see ["Roles"](#) chapter in "Settings" documentation.



In case of role change, you need to log out and log in. It might take up to 15 minutes for the current configuration to be effective depending on the system.

Onboarding and Configuring MindConnect Nano⁴

4.1 Overview of onboarding MindConnect Nano

This chapter describes the initial configuration of an asset and MindConnect Nano in order to onboard MindConnect Nano to Insights Hub. Onboarding is the process of attaching a MindConnect Element to Insights Hub. And also, it describes the process of connecting industrial data sources to Industrial IoT using MindConnect Nano.

To establish the initial connection between MindConnect Nano and Insights Hub (onboarding), you have to configure the essential asset and MindConnect Nano data. For onboarding the full configuration is not required, although you can accomplish all steps using Asset Manager. The respective data to be configured for onboarding is described in the following procedure.

Prerequisites

- MindConnect Element is connected to the Internet and powered on.
- The connection cable for the Internet is correctly plugged into the Ethernet port for the Internet (Industrial IoT).
- MindConnect Element is connected to asset or plant network.
- You have received the link to the Insights Hub UI with credentials (user and password data).
- A customer account is created.
- A standard USB device with a single partition in FAT or FAT32 is formatted for MindConnect Nano only.
- You have the role "mdsp:core:TenantAdmin". The user roles are set in Settings.



Prepare MindConnect Element

To prepare your MindConnect Element for V3, you have to update the firmware for boxes that were already connected to V2 (SAP). For devices connecting to Insights Hub region China 1, the MindConnect Nano firmware should be V3.3.0.4 or higher. You can find more information in chapter [Manual firmware update of MindConnect Nano](#).

Onboarding Procedure

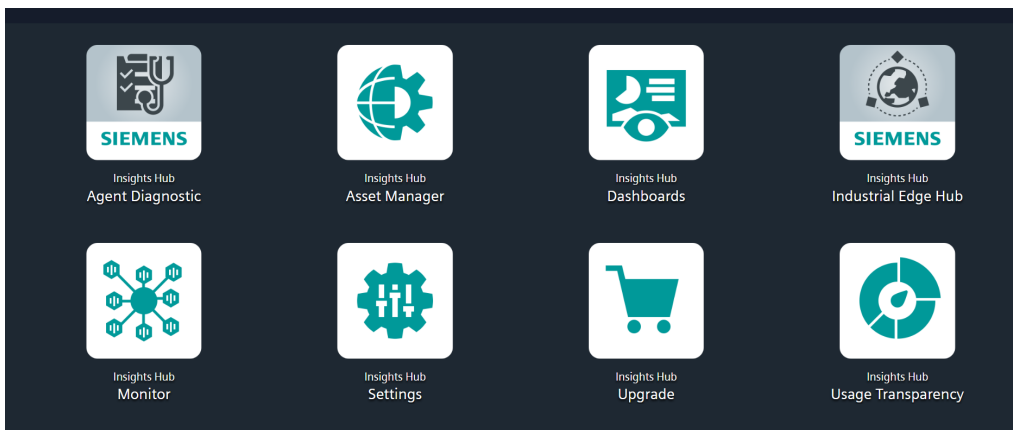
In order to onboard MindConnect Nano, follow these steps:

1. [Log in to Insights Hub](#)
2. [Creating an asset for MindConnect Nano](#)
3. [Enable connection to MindConnect Nano](#)
4. [Configuring network settings.](#)
5. [Transferring configuration to MindConnect Nano](#)

4.2 Logging in to Insights Hub

To log in to Insights Hub, follow these steps:

1. Click the link provided via mail by the Siemens AG.
 - The Insights Hub landing page for login will appear.
2. Log in to Insights Hub using the configured Authentication Provider (WebKey by default).
 - You will be redirected to your personal Launchpad.
 - Depending on your offering the Launchpad Icons will be displayed.



Insights Hub supports English, German and Chinese languages. Therefore, it is recommended to set the browser default to English, if you use a different language.

4.3 Creating an asset for MindConnect Nano


To connect your MindConnect Nano in Asset Manager, you need to create an asset. This process is the same for different types of MindConnect Elements that includes the Physical Asset type, and the Virtual Asset type.

For information on creating a Virtual Asset type, refer ["Creating a type in Asset Manager for MindConnect Nano"](#).

Procedure

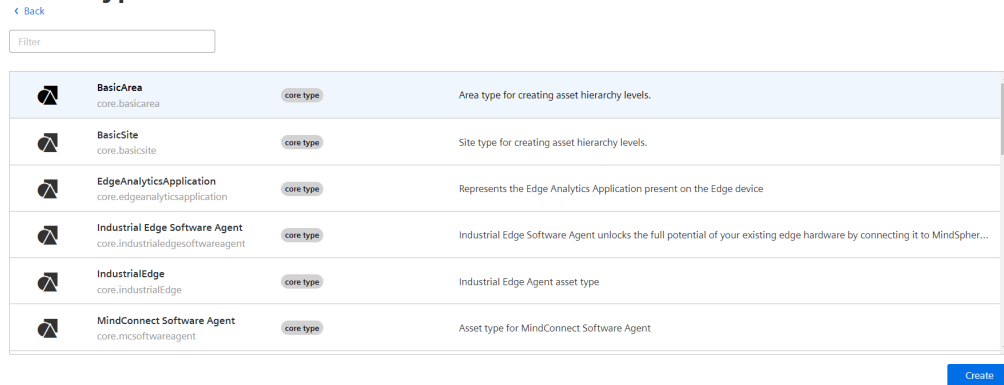
To create an asset, follow these steps:

1. Select Asset Manager from the Launchpad.
2. Click "Assets" in the navigation area.

3. To create a new asset click .

The asset type overview window is opened.

Select type



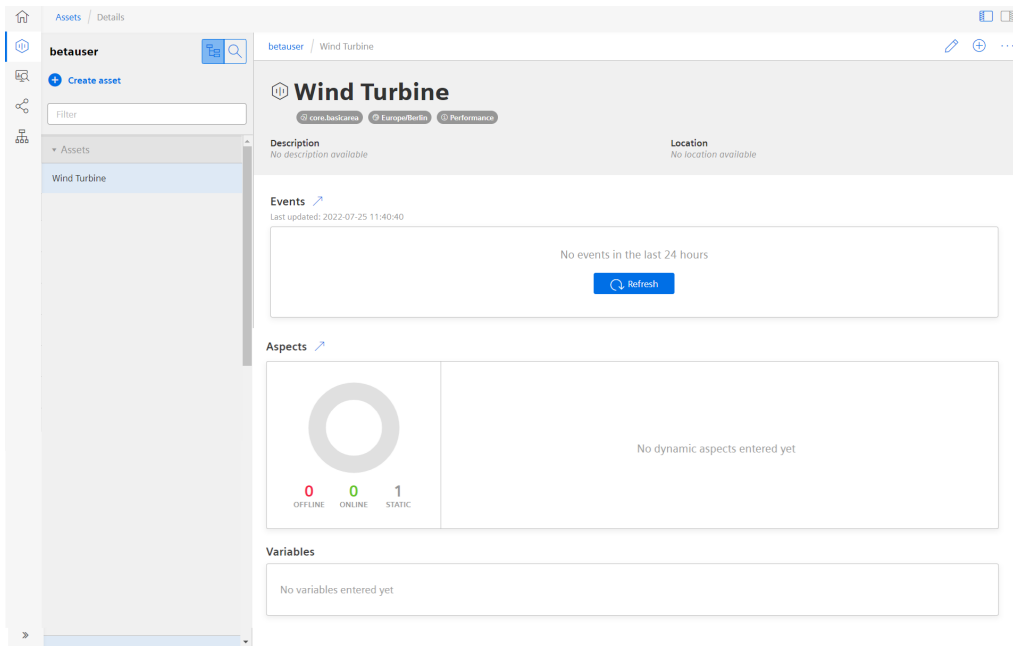
4. Select the respective asset type for your MindConnect Element, e. g. MindConnect Nano.
5. Enter a name and the data for the asset, e. g. "Wind turbine"
6. Confirm the entries with "Save".

Result

The new asset is available at the asset list.

The following graphic shows the created asset:

4.4 Enable connection to MindConnect Nano



4.4 Enable connection to MindConnect Nano

To use a MindConnect Nano with Industrial IoT you have to enable the connection. Within enabling the connection you can assign a MindConnect Nano to an asset.

Procedure

To enable the connection to MindConnect Nano, follow these steps:

1. Select the asset in the "Asset" tab.


In the "Connectivity" area click on the MindConnect Element plugin icon, e. g. "MindConnect Nano".

2. The "Create MindConnect" screen appears.

MindConnect Nano Plugin
MindConnect Nano
MindConnectNano

Create MindConnect

The unique ID can be found on your hardware. See image below as reference



Unique ID: *

Save

3. Enter the unique ID from your hardware.

Result

You have enabled the connection to the MindConnect Nano.

4.5 Configuring network settings

Network configuration of MindConnect Nano requires information on the ethernet interfaces for data acquisition (ethernet labeled with X2P1) and Industrial IoT (ethernet labeled with X1P1). If your company uses a proxy server, it must also be configured to connect MindConnect Nano to Industrial IoT. Contact your IT administrator for details on your local network settings. After enabling the connection to the MindConnect Element, you can edit the network configuration. You can enter either a static IP address or use DHCP. The last option is possible, if there is a DHCP server available in your company network.

Procedure

To configure the network setting, follow these steps:

4.5 Configuring network settings

1. Select the asset in the "Asset" tab.
2. Click the asset icon, e. g. "MindConnect Nano".
3. In the Agent settings section, the following graphic displays the edit MindConnect window.

MindConnect Nano
Here you can configure your MindConnect Nano to establish connectivity to Insights Hub.

CURRENT VIEW: [Agent Settings](#)

Device Identification

Country * Serial Number * Device ID

Please choose the country where the MindConnect Nano device is located. The serial number can be found on the local webserver. Created device ID reference.

Web Interface

☒ Use DHCP
Automatic IP assignment for your device

Production Interface

☒ Use DHCP
Automatic IP assignment for your device

Communication Settings

Proxy Type

Proxy protocol

Proxy Address

Proxy Authentication Type

Username

Password

Trigger Actions

- Download Onboarding Key
- Update Firmware
- Restart Device
- Offboard Agent
- Upload Log Files now

4. Enable "DHCP" if the DHCP server is available in a company network or enter the address information for static IP address.

The following graphic shows the IP address input fields:

Web Interface

☐ Use DHCP
Automatic IP assignment for your device

IPv4 Address Subnet Mask Gateway

DNS Server

Production Interface

☐ Use DHCP
Automatic IP assignment for your device

IPv4 Address Subnet Mask Gateway

DNS Server



- It is recommended to add network routes instead of using Gateway field. It is applicable for "Production Interface" only.
- Configuring "Production interface" is not necessary for the case of reading data from serial devices using Modbus RTU protocol and the setting DHCP is enabled by default.

5. Enter the proxy data area "Communication Settings" if necessary and click "Save".

- In this case, contact your network administrator to get the proxy data. For Proxy IP address input field, it is also possible to define the port number, for example: 127.0.0.1:8080.

- For proxy authentication type NTLM: The domain and the username have to be provided as "username@domain" or as "domain/username".



If nothing else is configured manually, MindConnect Nano will try to obtain IP addresses for both adapters via DHCP.


6. Enter the available router gateways used in your company network.

7. Click "Save" to complete the configuration.

4.6 Configuring diagnostic settings

Diagnostic Settings

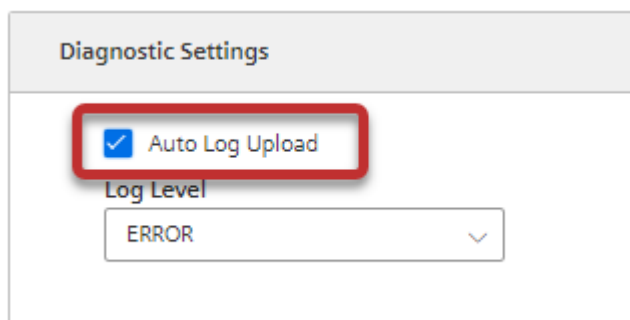
This section contains the configurations related with logging. To perform a diagnostic settings change, follow these steps:

1. In "Asset" tab, select the asset.
2. Open MindConnect plugin.
3. Click , next to "Onboarding Status".
4. In "Diagnostic Settings" section, make the changes in Auto Log Upload or Log Level.

5. Click "Save".

Auto Log Upload

If Auto Log Upload is enabled, the device uploads all the generated logs to Insights Hub periodically. Log upload functionality for the device is enabled by default.



Diagnostic Settings

☒ Auto Log Upload

Log Level

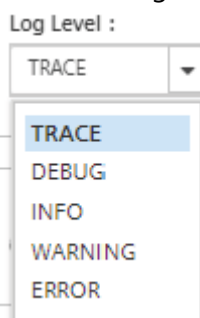
ERROR



In case the user does not wish to upload logs the checkmark can be unchecked and no logs would be uploaded further.

Log Levels

Log level represents the log level of MindConnect device. There are five possible log levels, and the hierarchy is as follows in lowest (TRACE) to highest (ERROR) priority. In “TRACE” mode, the device will log extensively and in “ERROR” mode device will only log error messages.



Log Level :

TRACE

TRACE

DEBUG

INFO

WARNING

ERROR

The selected log level and logging mechanism works as logging only selected level and the higher levels.

Log level is set to “INFO” by default.

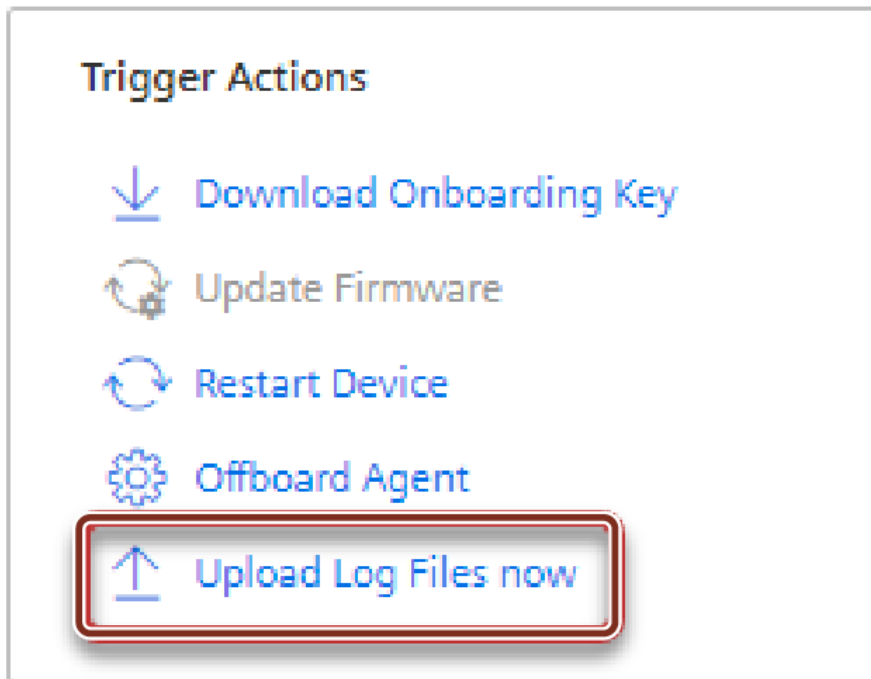


Auto Log Level Change

- “TRACE/DEBUG” log levels provide more verbose logs which will be helpful to troubleshoot issues, however they produce more files. If the log level is set to “TRACE/DEBUG”. It will be automatically changed to INFO after one day.
- Auto log level change operation will not be synchronized with the asset’s device configuration.

Upload Logs Now

If you click “Upload Logs Now”, the device immediately uploads latest logs to Insights Hub regardless Auto Log Upload status. It can be used for troubleshooting purposes.



Remote Services Integration

The integration of Remote Services to MindConnect Nano aims to provide solution for accessing factory related device via Nano. There are plenty of different services and protocols offered by RS to connect to the field devices and data. For more information on RS application, refer to [Remote Services](#) documentation.

To integrate RS application with MindConnect Nano, follow these steps:

1. Create asset.



To create an asset, see [Creating an asset for MindConnect Nano](#).

2. Onboard agent.

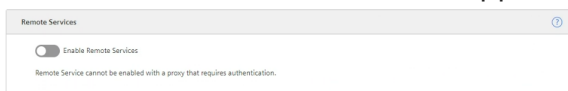


To onboard an agent, see [Overview of onboarding MindConnect Nano](#).

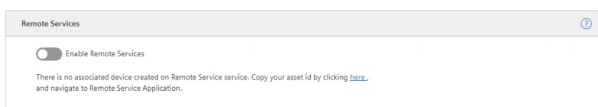
3. Link asset id with RS application.
4. Enable RS Services on the agent using toggle button and configure use cases in RS applications.
5. Click "Save".



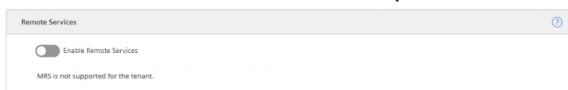
- Be aware that enabling RS application might effect on the time-series data transfer. Ensure that MindConnect Nano is able to perform operations smoothly via Insights Hub Monitor occasionally.
- RS Application does not support proxy authentication, thus enabling the proxy authentication will disable the RS application if it is enabled.



- Starting or stopping RS application is an asynchronous task, during this ongoing phase the toggle on or off operations will be neglected.
- RS application itself has a separate management system of devices, the user needs to create a device on RS application and associate it with MindConnect Nano's asset via their id's. Thereafter, user can able to enable the application from the MindConnect element UI.



- If the RS application is not provisioned for the customer environment then the user will not be able to enable the application and will see the below message on MindConnect element UI. To provision the application contact Support Team.



4.7 Configuring Buffer settings

Buffer Mechanism

In buffer mechanism, there are two options:

- Default Buffer Mechanism: It provides parallel upload from cache and real time data. It prioritizes newest data to old data while storing (works as a last in, first out (LIFO))
- Chronological Buffer Mechanism: It is chronological upload of data. If "Chronologic Upload" is selected, the buffer mechanism will prioritize oldest data to new data (works as first in, first

out (FIFO)).

You can configure the buffer behavior from "Network and Diagnostic Settings" screen.




If the buffer behavior is changed, it will restart the data collection process and this process will take few minutes.



Data loss

The data collection will be stopped during the time of buffer behavior change. The amount of data collection is stopped for few minutes, it depends on restart of the device. It may take longer time on slow devices like, IoT2040.

To perform a buffer behavior change proceed as follows:

1. In "Asset" tab, select the asset.
2. Open "MindConnect" plugin.
3. Click , next to "Onboarding Status".
4. In "Buffer Behavior" section, select "Chronological Upload".
5. Click "Save".

Buffer Behaviour

☒ Chronologic Upload

Save



Chronologic Upload Specification

- If there is a bad network connection or no connection at all, the data is going to be stored in cache memory. In "Chronologic Upload", overwritten of cache memory is not possible. Only if there is any space on cache memory, the data is going to be written to it. If the cache memory reaches out its limit and filled with data, then the data is not going to be saved on cache memory. It will cause data loss until some data is uploaded.
- As chronological upload the flow of real data to cloud will take longer time comparing to default behavior.
- The events are not cached in any case of the disconnection. Therefore, "Chronologic Upload" is not applicable for the event upload.

4.8 Transferring configuration to MindConnect Nano

Requirements

- A standard USB device with a single partition in FAT or FAT32 is formatted.
- You have enabled the connection to the MindConnect Nano.
- You have configured the network settings.

Procedure



USB stick information

The configuration created in this step includes security relevant information. You can encrypt and sign the information. Take care of the USB stick and the information flow from Industrial IoT to USB stick and from USB stick to the MindConnect Nano. Delete the USB stick after use.

To export the configuration, follow these steps:

1. In the MindConnect settings click .

The following graphic shows the "Edit MindConnect" screen:

MindConnect Nano

Device ID: f5f68c7b-9e41-4dbf-aebc-f203831319a0

FD953600 Offline Onboarding... Unknown

UID Connection Status Onboarding Status Firmware

Edit MindConnect

Offboard Download Onboarding Key

Configuration

Unique ID: *

FD953600

This ID can't be changed after the asset is onboarded!

Web Interface

Connect MindConnect with MindSphere

DHCP ☒

Production Interface

Connect MindConnect with your local production interface

DHCP ☒

Communication Settings

Proxy Type:

NONE

* required input field

Save Cancel

2. Click on "Download Onboarding Key".
3. Save the downloaded file to the root directory of your USB stick.
4. Insert the USB stick into the USB port of the MindConnect Nano, e. g. "MindConnect Nano"



During the import of the configuration, the following LED-states are to be expected:

1. MAINT LED flashes orange
 2. RUN LED blinks green
 3. On-boarding result
 - RUN LED green: MindConnect onboarded successfully
 - MAINT LED orange: onboarding failed
- The process can take up to 30 minutes. Please do not remove the USB stick during this time.

5. After successfully importing the configuration, remove the USB stick from the MindConnect Nano.
6. Check the status of the MindConnect Nano at the Overview.

Result

The connection status in Asset Manager at the overview has changed to "Onboarded". The configuration of the asset has been transferred to the MindConnect Nano. The connection between the MindConnect Nano and Industrial IoT is established.

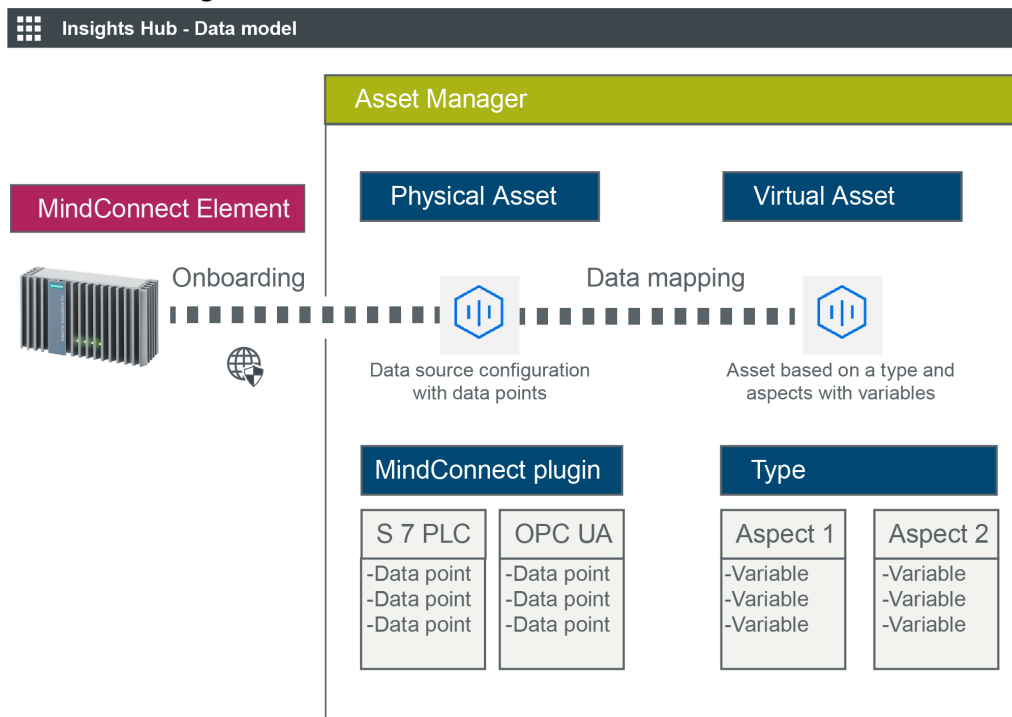
Once MindConnect Nano is onboarded, the connection to your asset is permanent and your asset for data collection can be configured. This requires configuration of machine data to be monitored. Any additional configuration (except network configuration) of the connected asset will automatically be synchronized with your onboarded MindConnect Nano.

The connection to the asset is permanent and can only be cancelled by offboarding. For this purpose, refer to the [Asset Manager](#) documentation.

4.9 Configuring data in Asset Manager for MindConnect Nano

After onboarding a MindConnect Nano, you need to configure the data in Asset Manager. You have to set up a data connection to receive data from your MindConnect Nano.

In order to use the data from your MindConnect Nano you need to map the received data to the Asset Manager data model:



Asset Manager uses the following areas:

- Assets
- Types
- Aspects

In order to configure data in Asset Manager you need to fulfill the following steps:

- [Creating an aspect and variables for MindConnect Nano](#)
In this step you create required aspects and variables.
- [Creating a type in Asset Manager for MindConnect Nano](#)
In this step you create an asset type to enable data mapping.

- [Adding a data source for MindConnect Nano](#)

The data source defines the reading of the data from the MindConnect Nano.

- [Mapping an aspect to a data source for MindConnect Nano](#)

In this final step you map the data of the MindConnect Nano to the aspects and variables.



In case of role changes or reassignment of roles, it might take up to 15 minutes for current configuration to be effective.

4.10 Creating an aspect and variables for MindConnect Nano

To use the data of your MindConnect Element you need to create aspects and variables in Asset Manager.


Aspects are combined, pre-configured data and form the context for the evaluation of industrial processes. An aspect can consist of several variables. Within the industry process, assets transfer the aspects as time series data.

In Asset Manager you can create aspect types. Aspect types are pre-configured templates for aspects. The advantage of aspects types is to use one type for several assets. For example, you can create one aspect type for all generators of a wind farm with the variable rotation speed.

For information on creating an asset, refer ["Creating an asset for MindConnect Nano"](#).

Procedure

To add an aspect type in Asset Manager, e. g. "Generator", follow these steps:

1. On the start screen click "Aspects".
2. To create a new aspect click .
3. Enter your Tenant ID for the aspect, e. g. "betauser.generator".



All custom aspects and types must be named as "<TenantID>.xxx"

4. Enter a name and description for the aspect e. g. "Generator". 5. Follow these steps for each variable, e. g. "Pressure" and "Temperature":

- Enter variable data.



- The units and data types specified here will need to match exactly(also case sensitive) with the physical data source.

Variables

Variable names must be unique inside an aspect.
The data type BIG_STRING is only available for a dynamic aspect.

[Add variable](#)
[Import variables](#)
[Download template](#)
[Export variables](#)
[Delete all new](#)

Name	Unit	Data type	Max. length	
Pressure	kPa	DOUBLE	Max. length	Defined
Temperature	°C	DOUBLE	Max. length	Defined

- Click "Add variable" to add other variables.
- 6.Click "Save".

Result

The new aspect type with its variables is available in the aspect list.

Library / Aspect Types / Details

Aspect Types

[Create aspect](#)

Filter

Generator
betasuser.Generator

Generator

Description
Generator

General Usages

Variables

Name	Unit	Data type	Max. length	Default value
Pressure	kPa	BOOLEAN	-	-
Temperature	°C	BOOLEAN	-	-

4.11 Creating a type in Asset Manager for MindConnect Nano

To enable data mapping, you first have to create an asset type. An asset type is a pre-configured template for an asset. The asset type predefines which aspects are integrated into the template.

You have to link aspects to an asset type to enable the connection to a data point.


For information on creating an asset, refer ["Creating an aspect and variables for MindConnect Nano"](#).

Requirements

- You have created an aspect.

Procedure

To create an asset type proceed as follows:

1. Click on the "Types" tab in Asset Manager.
2. Click on "BasicDevice".
3. To create a new asset type click .
4. Enter ID and name for asset type, e. g. "Wind turbine".

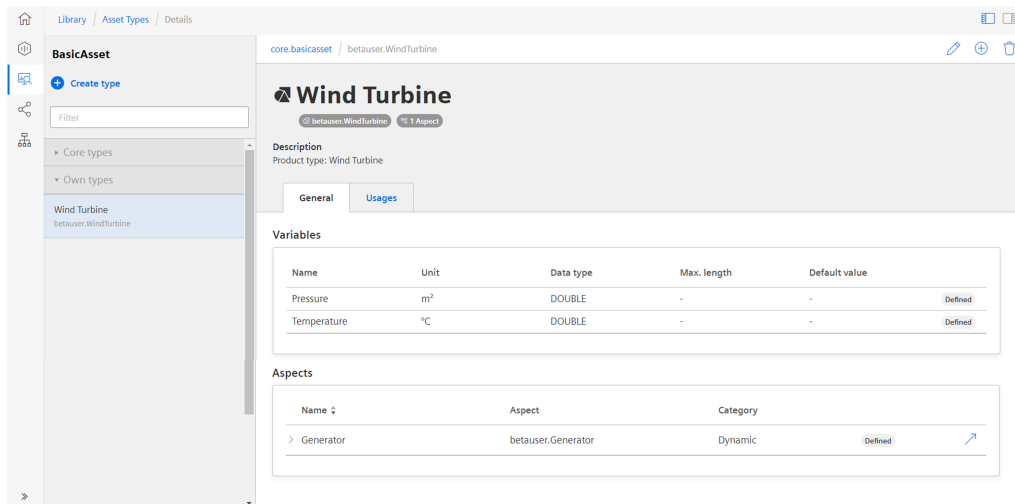


The ID must be named as "TenantID.xxx".

5. Click "Image" to add an image to the aspect type.(optional)
6. Click "Variables" to add variable(s) to the aspect type.(optional)
7. Click "Aspects" and click "Browse aspects" to select an aspect from the dropdown menu and then click "Add". For more information on adding aspect type. see ["Creating an aspect and variables for MindConnect Nano"](#).
8. Click "Save" to save the asset type .

Result

The new asset type is successfully created. The new asset type is now available in the presets. The following graphic shows the new available asset type "Wind turbine" in assets.



4.12 Adding a datasource for MindConnect Nano

To receive the data of your MindConnect Nano you need to add a datasource. The datasource specifies the protocol and reading cycle that is used to transfer the data from your asset to the MindConnect Nano. Within the datasource, you enter the address of the asset inside of your network.

Requirement

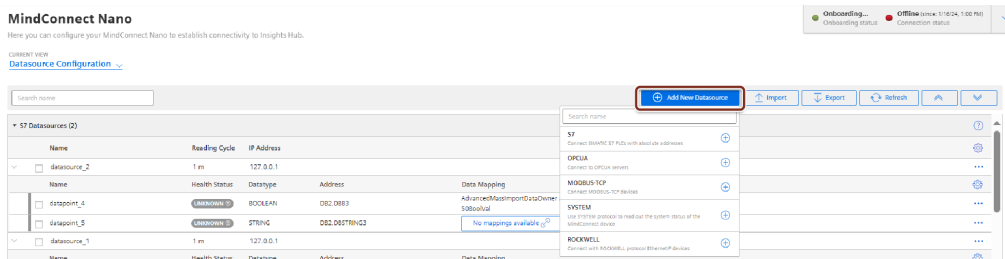
- You have created an asset and an aspect.
- You have enabled the connection to the MindConnect Nano.
- You have the IP address of the asset inside of your network.

Procedure

To add a datasource to an asset of type MindConnect, for example: "Generator", follow these steps:

1. Select the Asset in the "Assets" tab.
2. Click the asset icon, for example: "MindConnect Nano".

In the Data Configuration view, click "Add New Datasource" and select the datasource to
3. add a new datasource.



- The default reading cycle is 86400 seconds, which is one day.
- You can find additional information on supported protocols in the chapter [Overview of supported protocols for MindConnect Nano](#).

4. Enter the datasource parameter.



5. Click "Save".

Result

You have now created a new datasource. The new datasource is available at the asset details of your MindConnect Nano.

The following graphic shows the datasource added:



Cloning of Datasource

Clone of datasource creates a copy of datasource (including copy of datapoints). User needs to provide new name for datasource.

4.13 Adding datapoints for MindConnect Nano

A datapoint is a measurable value of an asset that can be represented numerically and graphically. Examples of a datapoint are temperature or pressure. You can add datapoints to a datasource to collect the data for example from a control unit. In the next step you have to link the datapoints of a datasource with the respective variables of an aspect.

You can find more information about how to send data in chapter [Send commands for MindConnect Nano](#).

Limitations

One datapoint cannot be mapped to two variables in the same aspect.

Health status

The health status shows the quality of the connection of the datapoint. You can click on the status to get further information. For more information about quality code, see [Quality code settings](#).

Requirement

- You have created a datasource.

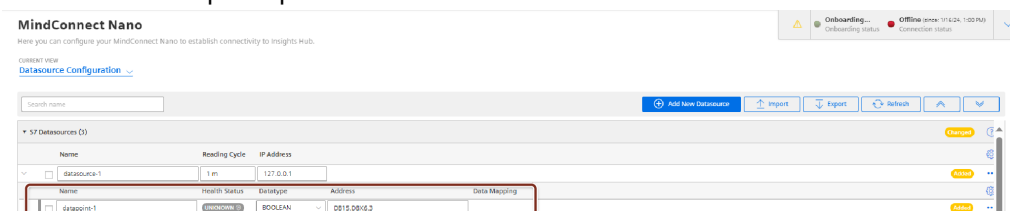
Procedure

To add a datapoint to a datasource, for example: "Pressure", follow these steps:

1. Select the asset in the "Assets" tab, for example: "Wind turbine".
2. Click the asset icon, for example: "MindConnect Nano".
3. Click "Add Datapoint" corresponding to the selected datasource.



4. Enter the datapoint parameters.



5. Click "Save".



- The Data types specified here will need to match exactly with the aspects and variables. If they are different, the aspect cannot be connected to the datapoint.
- The protocol datatypes specify the Datapoint Address. You can find additional information on protocol datatypes in the chapter [Overview of supported protocols for MindConnect Nano](#).
- "Data Acquisition Mode" defines the access type of datapoint. There are three acquisition types; "READ", "WRITE" and "READ&WRITE". To write data to a datapoint, "WRITE" or "READ&WRITE" type needs to be selected. Default value is "READ".
- It is possible to use hysteresis value in this field. This reduces the number of read events. The hysteresis filters signals and reduces the response to short-term signal fluctuations. The maximum value for hysteresis can be 99999 to minimum as 0.001.

Result

The new datapoint is available at the datasource.

In order to send the changes to the device, you have to click "Save" in the MindConnect plugin.

The following graphic shows the health status of the new datapoint:

MindConnect Nano
Here you can configure your MindConnect Nano to establish connectivity to Insights Hub.

CURRENT VIEW
[Datasource Configuration](#)

Search name ⊕ Add New Datasource ⬆ Import ⬆ Export 🔄 Refresh ⬆ ⬆

57 Datasources (3)				
Name	Reading Cycle	IP Address		
datasource-1	1 m	127.0.0.1		
Name	Health Status	Datatype	Address	Data Mapping
datapoint-1	🟢 Online	BOOLEAN	DB15 DBIN 3	No mappings available ⬆

Cloning of Datapoints

Clone of datapoint creates a copy of datapoint. The datapoint is cloned without a name(user needs to enter the name manually) and also without data mappings in case there are any existing data mappings for the datapoints.

4.14 Mapping an aspect to a datasource for MindConnect Nano

Data mapping means matching variables of an aspect with the respective data points of a datasource.

- Variables of an aspect represent the Industrial IoT data.
- Data points of a datasource represent the data of the MindConnect Nano.

To use the data of your MindConnect Nano you have to map them to the respective variables of an aspect.

Requirement

You have created an asset based on your desired aspect and asset type.



Browser compatibility

Microsoft Edge browser do have some issues with the user interface (example: "Mapping"). Use another browser like, "Chrome" or "Firefox".

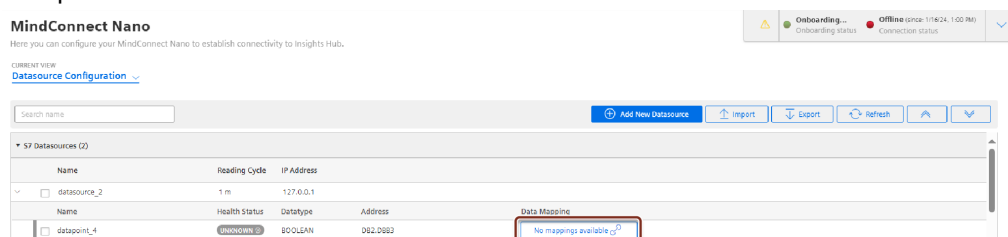
Data mapping can be done in following ways:

- [By using in-cell "No Mapping available" functionality.](#)
- [From "Datapoint Mapping" option.](#)

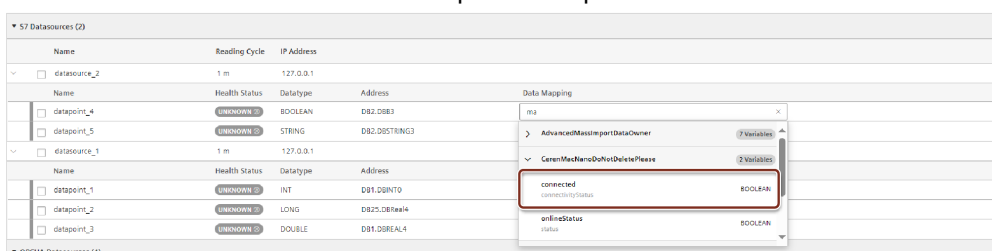
Map a variable to a datapoint by using in-cell functionality

To map a variable to a datapoint by using in-cell "No Mapping available" functionality, follow these steps:

1. Select the asset in the "Assets" tab, for example: "Wind turbine1".
2. Click the MindConnect Nano Plugin icon.
3. Click "No mapping available" to map a variable in the Data configuration view for the datapoints added.



4. Enter and select the variable to map the datapoint.



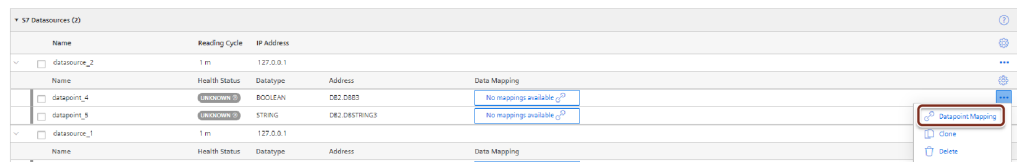


Only data types that match exactly with the aspects and variables are available. If they are different, the aspect can't be connected to the datapoint.

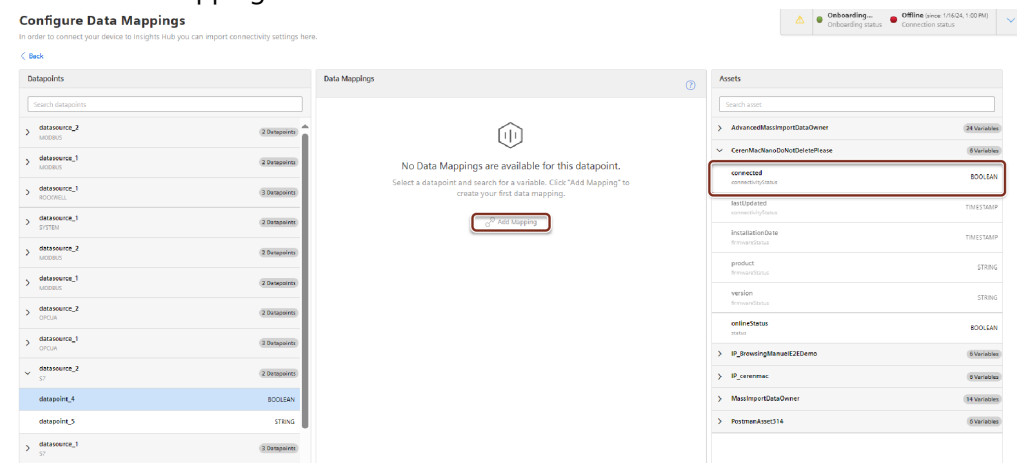
Map a variable to a data point from the "Datapoint Mapping" option

To map a variable to a data point by "Datapoint Mapping" option, follow these steps:

1. Select the asset in the "Assets" tab, for example: "Wind turbine1".
2. Click the MindConnect Nano Plugin icon.
3. Click "Datapoint Mapping" as shown in the image.

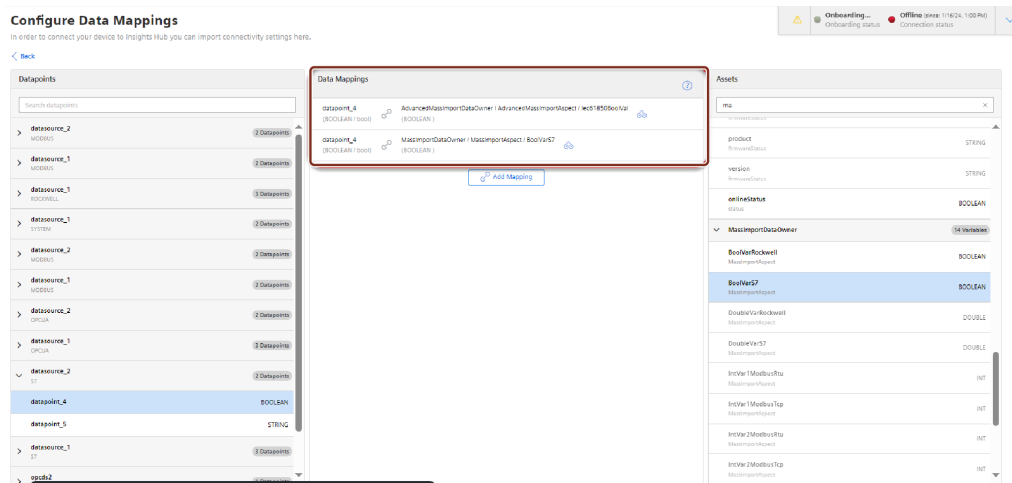


4. Enter the mapping variable in the Assets tab to select the variable and click "Add Mapping".



5. To map multiple datapoint, select another variable to be mapped and click "Add Mapping". The below show the multiple data mapping.

4.15 Browsing datapoint of a datasource for MindConnect Nano



Result

The datapoint is now mapped to the variable. User can also perform mappings for multiple datapoints.

The following graphic shows a data mapped variable:

Name	Reading Cycle	IP Address	
datasource_2	1 m	127.0.0.1	
Name	Health Status	Datatype	Address
datapoint_4	Unknown	BOOLEAN	DB2.DB83
datapoint_5	Unknown	STRING	DB2.DESTRING3

The following graphic shows a Multiple data mapped variable:

Name	Reading Cycle	IP Address	
datasource_2	1 m	127.0.0.1	
Name	Health Status	Datatype	Address
datapoint_4	Unknown	BOOLEAN	DB2.DB83

4.15 Browsing datapoint of a datasource for MindConnect Nano

After adding the datasource, you can browse the datapoints for the selected data source.



- This feature is only supported for OPC UA protocol.
- During configuration of data points via browsing feature, performance may get impacted depending on the configuration and data points in Data Sources.

Prerequisite

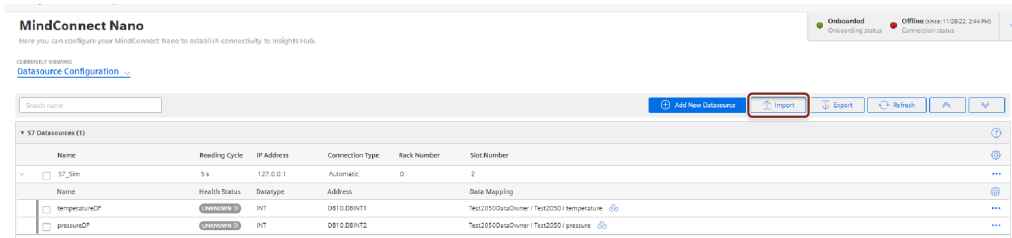
- You have created a datasource.


- After creating the datasource, click "Save" and "Apply changes".

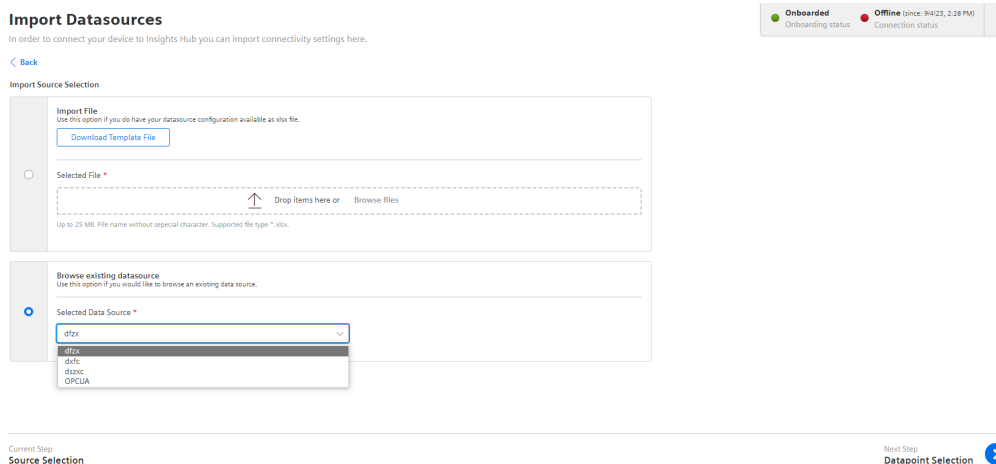
Procedure

To browse the datapoints of a datasource, follow these steps:

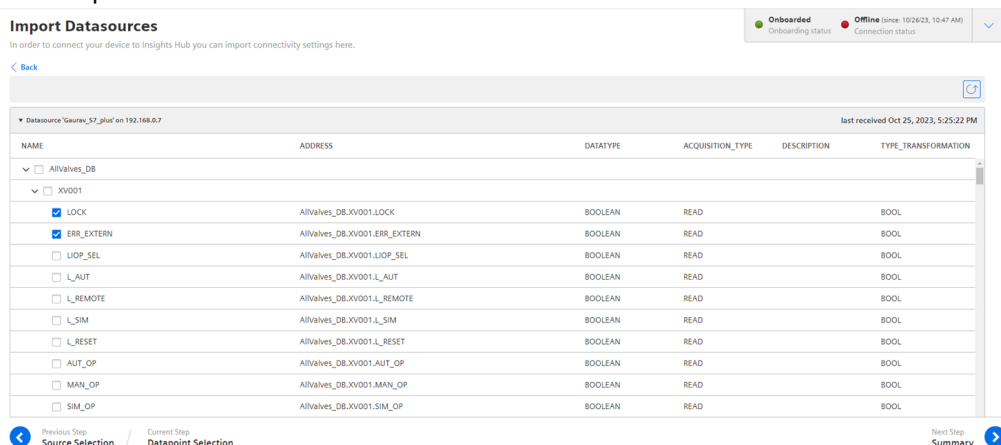
1. In the "Assets" tab, select the asset.
2. Click MindConnect Software Agent plugin.
3. In "Datasource Configuration" view, click "Import".




- In the "Browse existing datasource" tab, select the data source and click  to proceed to 4. datapoint selection step.



- After browsing the datasource to retrieve datapoints, navigate and select the datapoints to 5. be imported.



6. Click  to proceed to next step.
7. Click "Start Import" to browse the datapoints for the selected data source.

Preview of Imported Datasource configurations

S7+** Datasources

Total

1

including 2 datapoints

Import Configuration

☐

Overwrite existing configuration

All existing datasources and datapoints will be deleted. New datasources and datapoints will be created as imported.


☒

Add to existing configuration

New datasources and datapoints will be added. No existing resource will be deleted during import.

Start import

By clicking "Start Import" we are going to create the defined resources for you.

 For Browsing, Overwriting the existing configuration is not available.

Result

The available datapoints are displayed for the selected data source.

1

2

3

4

Available datapoints for s7plustest

Last Retrieval: 2022-06-15 13:36:19

Close

Name	Address	Datatype	Acquisition type	Description	Type Transformation
INTCOA_DB					
CHAR	L_CHAR	DOUBLE	READ&WRITE		USINT
Q_TRIANGLE	Q_TRIANGLE	LONG	READ&WRITE		INT
BOOL	BOOL	BOOLEAN	READ&WRITE		BOOL
BYTE	BYTE	DOUBLE	READ&WRITE		BYTE
CHAR	CHAR	DOUBLE	READ&WRITE		USINT
WORD	WORD	DOUBLE	READ&WRITE		WORD
INT_TRIANGLE	INT_TRIANGLE	LONG	READ&WRITE		INT
DWORD	DWORD	DOUBLE	READ&WRITE		DWORD
DINT	DINT	LONG	READ&WRITE		DINT
REAL_SINUS	REAL_SINUS	DOUBLE	READ&WRITE		REAL
RECTANGLE	RECTANGLE	LONG	READ&WRITE		INT
TIME	TIME	TIMESTAMP	READ&WRITE		TIME
TIMEOFDAY	TIMEOFDAY	TIMESTAMP	READ&WRITE		TIME_OF_DAY
DoubleDataDB					
dataDB500MS					
BoolDataDB					
EC_Time_0_DB					
Global_DB					

Accept

Cancel

0 datapoints selected

Clear All

5

- ① Check and uncheck to select the datapoints of a data source
 - ② List of the datapoints of the selected data source
 - ③ Re-loads the datapoints forcefully
 - ④ Displays the last retrieval of the datapoints
- Note:** The retrieved datapoints of a selected data source is available only for 24hrs.
- ⑤ Displays the number of selected datapoints

4.16 Importing datasource configurations for MindConnect Nano

To receive the data from MindConnect Nano, add a data source with corresponding datapoints. You can also upload a file for faster import of the configuration or download the template file.

Prerequisites

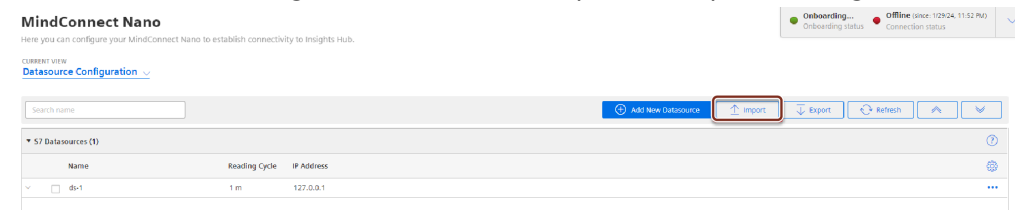
The prerequisites to configure the data sources are:


- Asset and aspect are created.
- MindConnect Nano is connected.
- Prepare a file to import according to the following guidelines.
- Check the structure in the template file. It differs for each plugin type.

Procedure

To import the datasource configuration for an asset of the type MindConnect Nano, follow these steps:

1. In the "Assets" tab, select the asset.
2. Click MindConnect Nano plugin. For example, "MindConnect Nano".
3. In "Datasource Configuration" view, click "Import" to import a configuration file.



4. In the "Import File" tab, browse the data source configuration file and click .

Import Datasources

In order to connect your device to Insights Hub you can import connectivity settings here.

[Back](#)

Import Source Selection

Import File

Use this option if you do have your datasource configuration available as .xlsx file.

Download Template File

Selected File *

Drop items here or

Browse files

Up to 25 MB. File name without special character. Supported file type *.xlsx.

Browse existing datasource

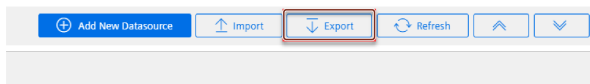
Use this option if you would like to browse an existing data source.

Selected Data Source *

Windturbine



- To check the format of the data configuration file before importing the configuration, download the template file by clicking download template.
- The template file consists of an example data configuration. In the below given screenshot, the information related to data source and datapoint declarations for S7 and it can switch to each protocol from the bottom of the file (e.g S7, OpcUa, modbus-tcp, etc).
- "Export" option makes the user to export the existing data source configuration of an asset, where the existing data source configuration is exported as file and can be imported to overwrite or add to the existing configuration.



- The supported protocols for import and export configurations are S7, OPC UA, Modbus-TCP, Modbus-RTU, Rockwell and System.
- For OPC UA Data source, configurations like OPC UA security certificates need to be added manually and are not supported via export and import.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	Tag	Datasource Name	Datasource Description	Read Cycle	Address	Connection Type	Bus	Slot	Datasource Name	Datapoint Description	Datatype	Unit	Address	Acquisition Mode	On Data Change	Hysteresis	Asset Name Mapping	Aspect Name Mapping	Variable Name Mapping
2	DS	datasource_1	datasource_description	1 m	127.0.0.1	Automatic													
3	DP								datapoint_1	datapoint_description	INT	num	DB1.DBB10	READ	TRUE		asset name1	aspectname1	variablename1
4	DP								datapoint_2	datapoint_description	LONG	num	DB1.DBB14	READWRITE	FALSE		asset name2	aspectname2	variablename2
5	DP								datapoint_3	datapoint_description	DOUBLE	num	DB1.DBB18	WRITE	FALSE		asset name3	aspectname3	variablename3
6	DS	datasource_3	datasource_description	1 m	127.0.0.1	Manual		0	5										
7	DP								datapoint_4	datapoint_description	BOOLEAN	bool	DB2.DBB3	READ	FALSE		asset name4	aspectname4	variablename4
8	DP								datapoint_5	datapoint_description	STRING	string	DB2.DBB3	READ	FALSE		asset name5	aspectname5	variablename5
9																			
10																			
11																			



- In the event of failed datasource validation, the import of datasource and the datapoints of datasource is skipped. However, if the error is on the datapoint level then the respective datapoint is skipped.

5.To delete and add new data configuration, select "Overwrite existing configuration" or to add new data configuration to the existing data configuration, select "Add to existing configuration".

Import Datasources

In order to connect your device to Insights Hub you can import connectivity settings here.

[< Back](#)

Preview of imported Datasource configurations

57 Datasources
Total

2
including 1 datapoints

Import Configuration



Overwrite existing configuration

All existing datasources and datapoints will be deleted. New datasources and datapoints will be created as imported.



Add to existing configuration

New datasources and datapoints will be added. No existing resource will be deleted during import.

Start import

By clicking "Start import" we are going to create the defined resources for you.

6.Click "Start import".



- Track the errors or warnings from "Import Configuration" or "Export Configuration".

- If the values for datasource Import and Export are not specified during import replace the values by default values. For the list of default values, refer to [Default values for datasource Import and Export](#).

Result

Data source configuration are imported with mapped datapoints. The new data sources is available at the asset details of MindConnect IoT2040. Datamapping is available at the corresponding datapoints.

Limitations of Import features

The limitations of Import feature are,

- Refreshing the browser page during import will interrupt the import process.
- Ensure reading cycle has **integer** values before importing.
- Double-check the hysteresis values as per the data type.
- For boolean fields, any values other than **true** will be imported as **false**.
- Non mandatory fields will be imported as **blank** in case fields are not specified.
- Validate that the Asset, Aspects and variable exists in the tenant before importing.

4.17 Configuring MindConnect Nano using API

AgentManagement API

You can configure MindConnect Nano via api calls with public API's that are offered from Industrial IoT. To update the data source configuration of AgentManagement api, refer to the endpoint "Data Source Configuration Operations" in [Agent Management Service – API Specification](#).

For more information on how to authenticate and how to get the information related with the MindConnect element, refer to [App Development](#).

This below information will show, how to structure the configuration part for the above api request for specific industrial protocols.

Protocol configurations

You can configure the device to AgentManagement using the below protocol specific configurations that needs to be configured in the api call.



In case the body structure is not as expected, the configuration might be rejected or may lead to corrupt configuration. Always verify the operation from MindConnect Element UI.

The json configuration in body part of the request consists of below key-value pairs. All the documented key-value pairs given below are important. The customData child fields of dataPoints and dataSource items are free formatted fields and should be a valid json. Below is the configuration body:

```
{
  "name": "DataSource Name",
  "customData": {
    "protocol": "Name of the protocol",
    "protocolData": { },
    "readCycleInSeconds": "Value"
  },
  "description": "",
  "dataPoints": [
    {
      "id": "a37d9e1fee6b4",
      "name": "DatapointName",
      "description": "-",
      "unit": "Unit",
      "type": "Type",
      "customData": {
        "dataPointData": { }
      }
    }
  ]
}
```



The above key-value pairs are important for the body, if missing any key-value pair will result in configuration failure.

Each protocol has its unique `protocolData` representing its protocol specific fields and `dataPointData` field for representing its unique datapoint configuration. In order to get the successful configuration for each one of the protocol, the free formatted parts of the body needs to be a valid json. Otherwise, the configuration will be failed.

The below given an example of how to configure a datasource with one datapoint for S7 protocol:



- The below json content can be changed as required, do not depend on it strictly.
- The free formatted fields "protocolData" and "dataPointData" has a specific configuration to S7 protocol.

```
{
  "name": "S7DS",
```

```

    "customData": {
      "protocol": "S7",
      "protocolData": {
        "ipAddress": "192.168.0.1",
        "manualRackAndSlot": "Automatic",
        "rackNumber": "0",
        "slotNumber": "2"
      },
      "readCycleInSeconds": "1"
    },
    "description": "",
    "dataPoints": [
      {
        "id": "a37d9e1fee6b4",
        "name": "DatapointName",
        "description": "-",
        "unit": "%",
        "type": "LONG",
        "customData": {
          "dataPointData": {
            "address": "DB1.DBWORD0",
            "hysteresis": "0",
            "onDataChanged": false,
            "acquisitionType": "WRITE"
          }
        }
      }
    ]
  }
}

```

Getting Configuration Details of Protocols

To get the configurations details for other protocols such as Modbus, Ethernet IP etc, the API of AgentManagement can be used to get the configuration details. The initial configuration can be done via UI and then using the below API config, it can be extracted and used for API calls and later for structuring.



The json content of each protocol varies and should be extracted after an initial UI configuration.

To get the data source configuration of AgentManagement api, refer to the endpoint "Data Source Configuration Operations" in [Agent Management Service – API Specification](#).

Configuring protocols

5

5.1 Overview of supported protocols for MindConnect Nano

The following are the supported protocols for MindConnect Nano:

- S7
- OPC UA
- Modbus TCP IP
- Modbus RTU
- System
- Rockwell

5.2 Overview of S7 protocol for MindConnect Nano

The following image shows the data source input window for the S7 protocol:

The screenshot shows the 'MindConnect Nano' interface. At the top, there's a status bar with 'Onboarded' (green dot) and 'Offline' (red dot) indicators. Below this, the 'CURRENTLY VIEWING' section shows 'Datasource Configuration'. A search bar is present. Below the search bar, there's a table titled 'S7 Datasources (3)'. The table has columns: Name, Reading Cycle, IP Address, Connection Type, Rack Number, and Slot Number. One entry is visible: 'datasource' with a reading cycle of '1 m', IP address '127.0.0.1', connection type 'Automatic', rack number '0', and slot number '2'. There are buttons for 'Add New Datasource', 'Import', 'Export', 'Refresh', and 'Change'.

Name	Reading Cycle	IP Address	Connection Type	Rack Number	Slot Number
datasource	1 m	127.0.0.1	Automatic	0	2

You can choose between automatic and manual connection type.

- Automatic: This mode tries to find slot and rack number automatically.
- Manual: In this mode you can enter the rack and slot number manually.

For S7 connection type the PG type-programmer-type-connection is used thus you will not be able to make use of PG connection.



For S7 300 : 400 PLCs, CPU can be dynamically located in any slot in the rack, in this case you can use manual configuration to locate CPU instead of auto find.

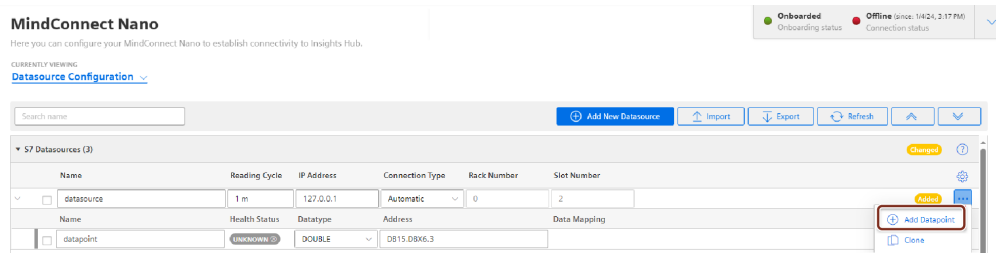
Symbolic access

This solution does not allow full symbolic access.

S7 data point parameter

To collect the data of your S7 device, click "Add Datapoint" to add new datapoint.

The following image shows the data point parameter window:



Requirements for S7-1200 and S7-1500

- Put/Get communication must be permitted (can be found e.g. S7-1500 -> Properties -> Protection & Security -> Connection mechanisms).
- The "Optimized block access" of the data blocks that you want to read must be deactivated (can be found in the data block properties).

Address format examples

Detailed examples:

Example address format DB15.DBX6.3:

DB15.DBX6.3

Area indicator			Data type	Offset address		
DB15	.	DB	X	6	.	3
Datablock address	Separator	Datablock	Bool	Offset Byte	Separator	Offset Bit
Reads 3rd bit of byte 6 of datablock 15						

Example address format IW10:

IW10

Area indicator	Data type	Offset address
I	W	10
Process Input	Word	Offset Byte
Reads a word of input starting from offset 10		

Overview

If you want to read data out of other areas, please replace “DB15.DB” from the examples with the area indicator from the list below:

Attention: only certain address examples work with Inputs and Outputs (see²⁾).

S7		Industrial IoT		
Data type	Length/unsigned/signed	Data type (length)	Address example	Description
Bool	1 bit	BOOLEAN (1 bit)	DB15.DBX6.3 ²⁾	Reads 3rd bit of byte 6 of datablock 15
Byte USInt	1 byte / unsigned	INT (4 bytes) LONG (8 bytes) DOUBLE *(8 bytes)* ¹⁾	DB15.DBB4 ²⁾ DB15.DBBYTE4	Reads an unsigned byte starting from offset 4 out of datablock 15.
SInt	1 byte / signed	INT (4 bytes) LONG (8 bytes) DOUBLE *(8 bytes)* ¹⁾	DB15.DBB4 ²⁾ DB15.DBBYTE4	Reads an unsigned byte starting from offset 4 out of datablock 15.
Char	1 byte / signed	INT (4 bytes) LONG (8 bytes) DOUBLE *(8 bytes)* ¹⁾	DB15.DBCHAR6	Reads a char starting from offset 6 of datablock 15.

S7		Industrial IoT		
Word	2 bytes / unsigned	INT (4 bytes) LONG (8 bytes) DOUBLE *(8 bytes)* ¹⁾	DB15.DBW10 ²⁾	Reads an unsigned word
UInt	2 bytes / unsigned	INT (4 bytes) LONG (8 bytes) DOUBLE *(8 bytes)* ¹⁾	DB15.DBWORD10	starting from offset 10 of datablock 15.
Int	2 bytes / signed	INT (4 bytes) LONG (8 bytes) DOUBLE *(8 bytes)* ¹⁾	DB15.DBINT12	Reads a signed integer starting from offset 12 out of datablock 15.
DINT	4 bytes / signed	INT (4 bytes) LONG (8 bytes) DOUBLE *(8 bytes)* ¹⁾	DB15.DBDINT28	Reads a signed double integer starting from offset 28 of datablock 15.
DWord	4 bytes / unsigned	LONG (8 bytes)	DB15.DBDW24 ²⁾	Reads an unsigned double
UDInt	4 bytes / unsigned	DOUBLE *(8 bytes)* ¹⁾	DB15.DBDWORD24	word starting from offset 24 of datablock 15.
Real	4 bytes / signed	DOUBLE (8 bytes)	DB15.DBD32 ²⁾	Reads a floating-point
			DB15.DBREAL32	number starting from offset 32 of datablock 15.

S7		Industrial IoT		
String	Length of variable, max. 254 bytes	STRING (max. 254 bytes)	DB15.DBSTRING10	<p>The maximum string length (Byte10) is read first. Then the number of bytes that Byte10 indicates are number read. After that the of bytes that Byte11 indicates are extracted from the byte array that is the result of second read operation.</p> <p>For example:</p> <ul style="list-style-type: none"> - (Byte10)-30 - (Byte11)-4 - (Byte12-Byte15)-TEST <p>First read result: 30 Second read result: 4TEST We will extract TEST from the second read result by the second read result by considering Byte11's value.</p>
String	Min. 100 bytes , max. 254 bytes	STRING (max. 254 bytes)	DB15.DBSTRING10,100	Reads 100 bytes of string starting from offset 10.

¹⁾ When DOUBLE is used for integers, they are converted to a floating-point number.

²⁾ For the areas Inputs and Outputs only certain address examples work.



Signed values

If you want to read a signed value, you have to use the respective signed datablock.

Example:

DB1831.DBW508 is unsigned. You can use DB1831.DBINT508 instead.



Real data type

If you want to write a value to REAL data type, you have to use "REAL" instead of "DBD" in the address.

Example:

DB15.DBD32 or DB15.REAL32 can be used for reading. You can use DB15.REAL32 for writing.

Area Types

S7 datapoint addresses must contain an area type.

Area Indicator	Area types	Description
C	Counter	Reads Counter value of hardware.
DB	Datablock	Reads datablock value from hardware.
DI	Instance Data	Reads data directly from hardware.
I	Process Input	Reads Input value of Process image.
M	Memory	Reads Memory value of hardware.
PI	Peripheral Input	Reads Input value directly from hardware.
PQ	Peripheral Output	Reads Output value directly from hardware.
Q	Process Output	Reads Input value of Process image.
T	Timer	Reads Timer value of hardware.

Area Number

S7 datapoint addresses must contain area number.

E.g.: DB10 (10. datablock)

Data Types

The following table shows the data types:

Data type	Length	Date range
BOOLEAN	1 bit	True to False
INT	4 bytes	-2147483648 to 2147483647
LONG	8 bytes	-9223372036854775808 to 9223372036854775807
DOUBLE	8 bytes	-9.99999999999999999999999999999999E+125 to 9.99999999999999999999999999999999E+125
STRING	1-255 bytes	-9.99999999999999999999999999999999E+125 to 9.99999999999999999999999999999999E+125

The following table shows the S7 data types:

S7 data types:

Data type	Length	Date range
-----------	--------	------------

Data type	Length	Date range
Bool	1 bit	True to False
Byte (unsigned)	1 byte	0 to 255
Word (unsigned)	2 bytes	0 to 65.535
DWord (unsigned)	4 bytes	0 to 4294967295
Char (signed)	1 byte	-128 to 127
Int (signed)	2 bytes	-32768 to 32767
DINT (signed)	4 bytes	-2147483648 to 2147483647
Real	4 bytes	+/-1,18e-38 to +/-3.40e+38
String	1- 254 bytes	+/-1,18e-38 to +/-3.40e+38

Offset

Variable types must have an offset. Offset is the starting byte number.

E.g. DW4 (a double word starting from byte 4)

String

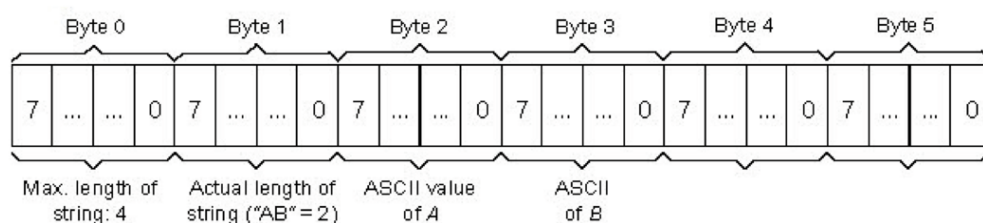
Data type	Length in bytes	Format
STRING[n] or STRING	n+2	ASCII character of any length. n specifies the length of the character string. A maximum length of 254 characters is permitted. If no length is specified, the default setting is 254 characters.

Data type	Examples of format used
STRING[2]	'AB'
STRING[55]	'The character string can consist of up to 55 characters'



You must enclose your character string in single quotation marks.

The following example shows the byte order when specifying the datatype STRING[4] with the output value 'AB'.



* Tod

TIME_OF_DAY (Time)	32	Time is in steps of 1 ms	TOD#0.0.0.0 to TOD#23:59:59.999	L TOD#1:10:3.3 L TIME_OF_DAY#1:10:3.3
--------------------	----	--------------------------	---------------------------------	------------------------------------------

* Time

Returns milliseconds.

Format:

The DT (DATE_AND_TIME) datatype saves the information on date and time of day in BCD format.

The following table shows the properties of datatype DT:

Length (bytes)	Format	Range of values	Example of value input
8	Date and time (year-month-day)		

5.3 Overview of OPC UA protocol for MindConnect Nano

OPC UA data parameter

The following image shows the data source input window for the OPC UA protocol:



For the OPCUA protocol, the data source input supports the following parameters:

Parameter	Description
Protocol	Shows the selected protocol type
Name	Name of the data source
Description	Add a description for data source
Reading Cycle	Enter a reading cycle for the data source

Parameter	Description
OPC UA Server Name	Enter the name of the OPC UA Server Name
OPC UA Server Address	Enter the OPC UA Server Address
OPC UA Server IP Address	Enter the OPC UA Server IP Address
OPC UA Authentication Type	You can choose between the following types: - NONE: No authentication enabled - BASIC: Enables the OPC UA BASIC authentication
OPC UA Security Mode	You can choose between the following modes: - NONE: No security mode - CERTIFICATE: Enables to upload a certificate in the following formats: *.pfx, *.pf12, *.cer, *.crt, *.der, *.p7b, *.p7r, *.spc
Enable OPC UA Events	You can enable the transfer and takeover of events and alarms in the OPC UA protocol to Industrial IoT. Insights Hub Monitor shows all events and alarms in the "Events" extension of the agent asset. Limits: - For Nanobox the proposed limit is 25 events/alarms per second - For IoT2040 the proposed limit is 10 events/alarms per second

OPC UA Data Access Specification

OPC UA Data Access Specification does not specify a certain representation for OPC UA Addresses (NodeId). A NodeId is represented by a combination of:

- NamespaceIndex
- IdentifierType
- Identifier

OPC UA Client Driver addresses have the following syntax: ns=;=.



OPC browsing is not supported.

For more information refer to the table below:

Field	Description
-------	-------------

Field	Description
NamespaceIndex	The index of the OPC UA Server namespace in which the address resides. If the index is 0, the entire ns=;= clause is omitted.
IdentifierType	The type of address. OPC UA supports the following four address types: i: A numeric address represented with 32-bit unsigned integer s: A string address containing characters that can be encoded using UTF-8 g: A GUID address in the format of XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX b: An opaque address (such as a byte string)
Identifier	The address that is formatted as a string. This address may be numeric, string, GUID, or opaque.

Examples NodeId

Below you can see examples of an OPC UA Addresses (NodeId) representation:

String:

String IdentifierType	Example
ns=[id];s=[string]	ns=5;s=Counter1

Attribute NodeId	Value NodeId
NamespaceIndex	5
IdentifierType	String
Identifier	Counter1



String for Identifier

The string you enter at the four Identifier will be case sensitive.

Numeric:

Numeric IdentifierType	Example
ns=[id];i=[number]	ns=0;i=3190 OR

Attribute NodeId	Value NodeId
NamespaceIndex	0
IdentifierType	Numeric
Identifier	3190 [BaseEventType_LocalTime]

Guid:

Guid IdentifierType	Example
ns=[id];g=[guid]	ns=1;g=0d4e10fa-b9e1-4842-bca6-887c7d8c31f0

Base64(Opaque):

Base64(Opaque) IdentifierType	Example
ns=[id];b=[base64]	ns=2;b=M/RbKBsRVkePCePcx24oRA==

Datatypes Conversion

The following table shows the Datatype Conversion:

IEC61131 elementary datatypes	OPC UA built-in datatypes	Industrial IoT Type
BOOL	Boolean	boolean
SINT	SByte	int
USINT	Byte	int
INT	Int16	int
UINT	UInt16	int
DINT	Int32	int
UDINT	UInt32	int
LINT	Int64	long
ULINT	UInt64	long
BYTE	Byte	int
WORD	UInt16	int
DWORD	UInt32	int
LWORD	UInt64	long
REAL	Float	double
LREAL	Double	double
STRING	String	string

IEC61131 elementary datatypes	OPC UA built-in datatypes	Industrial IoT Type
CHAR	Byte	string
WSTRING	String	string
WCHAR	UInt16	int
DT DATE_AND_TIME	DateTime	n/a
DATE	DateTime	n/a
TOD TIME_OF_DAY	DateTime	n/a
TIME	Double	double



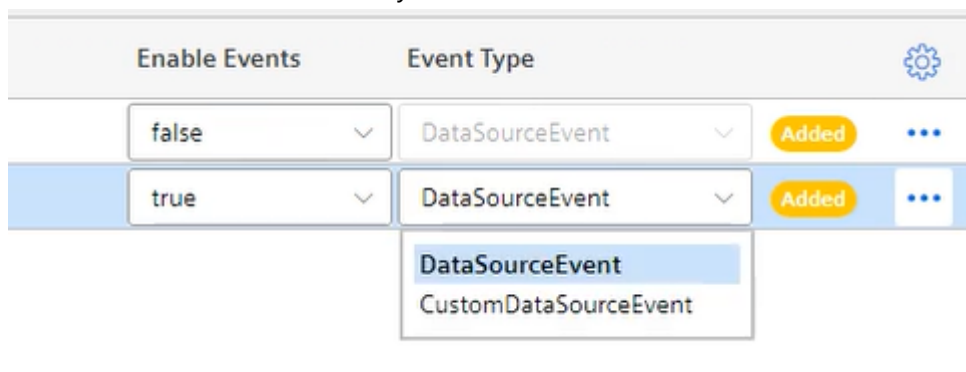
“NodeID” is not a valid part of the OPC string anymore.

OPC UA Event Type

You can choose default event type or one of the custom event types created in environment.

OPC UA Event Types

OPC UA plugins supports custom event types to be uploaded to Industrial IoT. Select the correct event type from the below drop-down menu, to get those events on Industrial IoT. In the dropdown menu, the events owned by the specific environment and the default event type “DataSourceEvent” is shown only.



The event types are summarized as below:

Data Source Event Type	It is the default event type which can collect some static fields of events from OPC UA server.
Custom Event Type	These types can be created by user/owner of environment, and they can collect both static and extra fields from OPC UA server.

A new event type must be created from "AgentBaseEvent" type with required extra fields and then, it can be selected from the event drop-down list which is shown during OPC UA server configuration.



- If there is no created custom event type in environment, "Data Source Event Type" will be selected by default.
- Both, data source events and custom events will be uploaded to Industrial IoT. They can be displayed on Insights Hub Monitor regarding to their types. For more information on displaying events, refer to [Introduction to Events](#) in Insights Hub Monitor documentation.
- There is a limitation on event exchange to cloud platform by 3k per minute and 20K per hour from the service Event Management.

Create custom event type via API

It is possible to create custom event types with extra fields via Event Management API. For more information on the service, see [Event Management - API Overview](#).

Below is the api to call using a technical user `/api/eventmanagement/v3/eventTypes`.

The following template shows the payload of the call:

```
{
  "name": "SomeCustomEventType",
  "parentId": "core.connectivity.event.type.AgentBaseEvent",
  "ttl": 35,
  "scope": "LOCAL",
  "fields": [
    {
      "name": "Message",
      "filterable": false,
      "required": false,
      "updatable": true,
      "type": "STRING"
    },
    {
      "name": "testField1",
      "filterable": false,
      "required": false,
      "updatable": true,
      "type": "STRING"
    }
  ]
}
```

```
    },  
    {  
      "name": "testField2",  
      "filterable": false,  
      "required": false,  
      "updatable": true,  
      "type": "STRING"  
    },  
    {  
      "name": "testField3",  
      "filterable": false,  
      "required": false,  
      "updatable": true,  
      "type": "STRING"  
    }  
  ]  
}
```



- The “name” field in body must be same as the “browseName” of the field which will be collected in the OPC UA Server. It is case-sensitive.
- Name for "Event Type" needs to be unique
- The “type” field must be “STRING” for all data types.
- The “scope” field should always be given as “Local” unless a particular reason is required to make it "Public".
- The field with name “Message” is the static field and it is not included to custom event types. It is needed to be added as an extra field.
- The fields Time, Severity and SourceName in the Event type are reserved fields and should not be created as extra fields.

OPCUA Browsing

OPCUA browsing feature enables the browsing of nodes on an OPC server for configuring the MindConnect Elements in an automated procedure.

Limitations

Unsupported data types in the browsing results are listed below, the nodes with

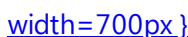
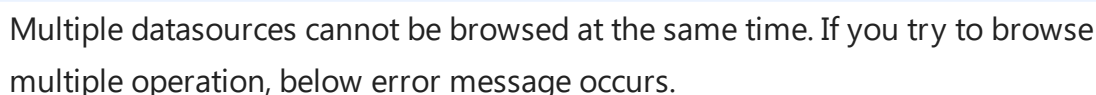
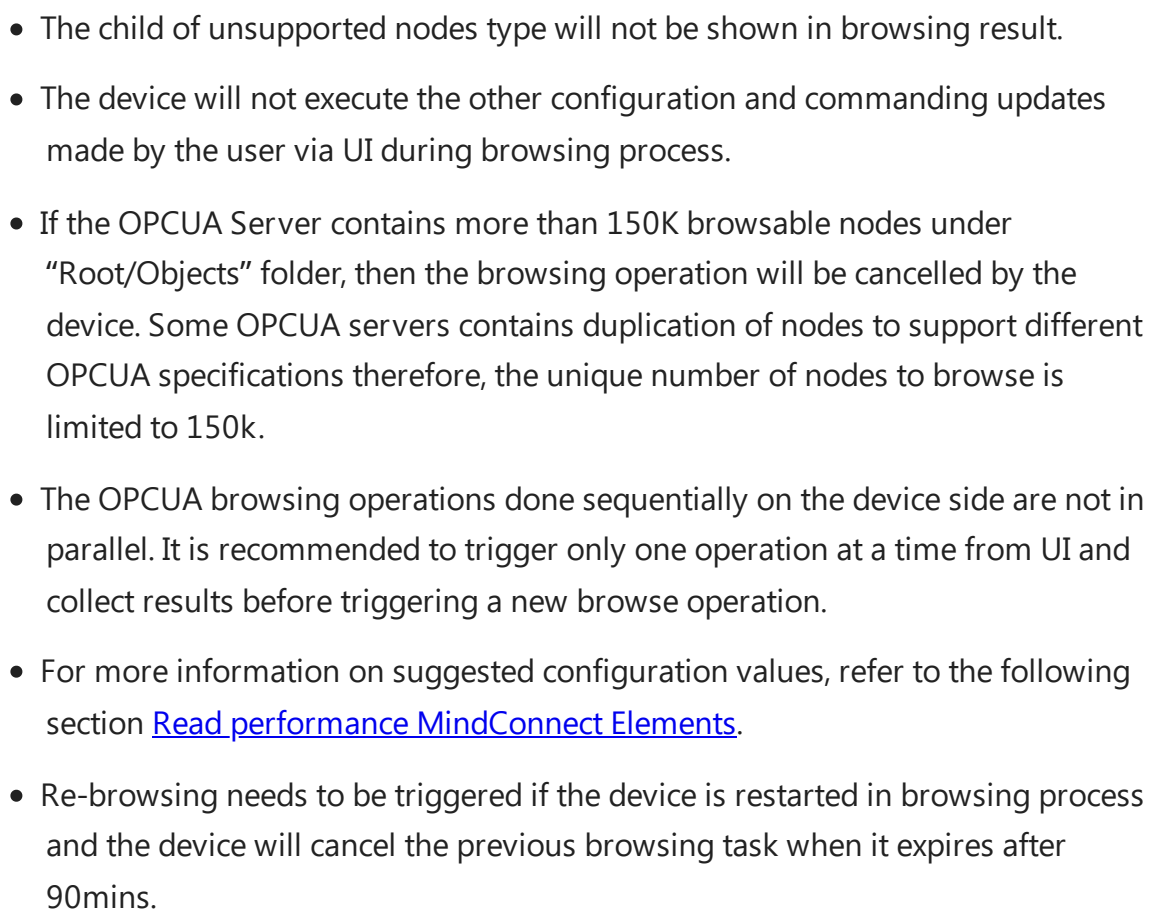
- unsupported types will not be shown in the browsing result.
 - Image, ImageBMP, ImageGIF, ImageJPG, ImagePNG
 - Enumeration
 - Number
 - Structure
 - Argument
 - ByteString
 - Range
 - EuInformation

If the time consumption is more than 90 minutes for browsing a datasource then the

- operation would be cancelled by the device.

If the OPCUA Server has more than 150K browsable nodes under 'Root/Objects' folder,

- then the browsing operation will be cancelled by the device.
- The browsing result of a datasource will expire after a day.
- The OPCUA browsing operations are done sequentially on the device side not in parallel.



!!! Warning Configuration jobs can only be applied after browsing operation is completed. If configuration job is applied during a browsing operation, below warning message is displayed on data source configuration page.

Type Conversion

Conversions of node supported types to Industrial IoT are mentioned in below table, if the OPC Type is not included in the below table, then that would be converted to STRING.

OPC Type	Cloud Type
Boolean	BOOLEAN
Float	DOUBLE
Double	DOUBLE
Duration	DOUBLE
SByte	INT
Byte	INT
Int16	INT
Int32	INT
Integer	INT
UInt16	INT
Int64	LONG
UInt32	LONG
UInt64	STRING
String	STRING

If the connection to OPCUA Server is disconnected during browsing, then the whole operation of browsing is failed. It is expected from the customer to trigger the browsing

- result again in those situations.

If a node with supported node type is not fulfilling the required fields on server side which is browsed than this node will be excluded, and it will not affect the whole browsing flow.

- The excluded node and its child nodes will not be displayed in browsing results.
- The default type for "Acquisition type" for browse result will be READ.

Performance of OPCUA Browsing

The browsing of an OPCUA server can be a time-consuming task depending on the number of existing nodes on the server. Below table describes the time consumption for idle devices which only perform browsing operations to OPC servers.

Number of Nodes	Type of Device	Time Consumed
5K	MindConnect Nano	~6 min
75K	MindConnect Nano	~16 min
150K	MindConnect Nano	~36 min



The reading cycles of datasources can be affected during the OPCUA browsing operation, as it requires a huge performance activity on the MindConnect Element.

5.4 Overview of Modbus TCP protocol for MindConnect Nano

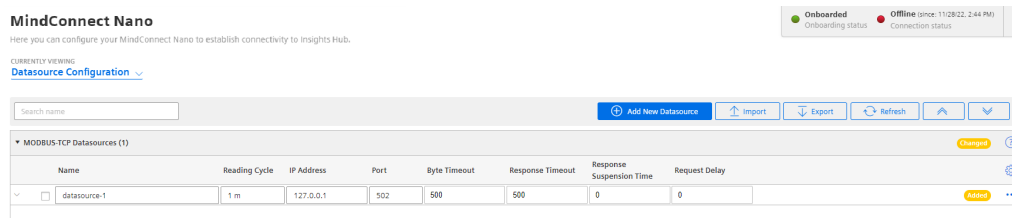
The Modbus protocol is supported for the MindConnect Nano plugin. You can select the protocol type Modbus TCP while adding a new data source in Asset Manager.



Modbus configuration

The Modbus configuration view is only visible after a successful onboarding of the MindConnect Nano. The MindConnect Nano Version must be higher than V3.3.0.2.

The following picture shows the protocol selection in Asset Manager:



For more information about how to create a new data source see chapter [Adding a data source and data points for MindConnect Nano](#).

Modbus TCP data source parameter

To connect a Modbus TCP device, you need to select the Modbus TCP protocol and enter the following data source parameters:

Parameter	Description
IP Address	IP address of the Modbus device
Port	Port number of the Modbus device (default: 502)

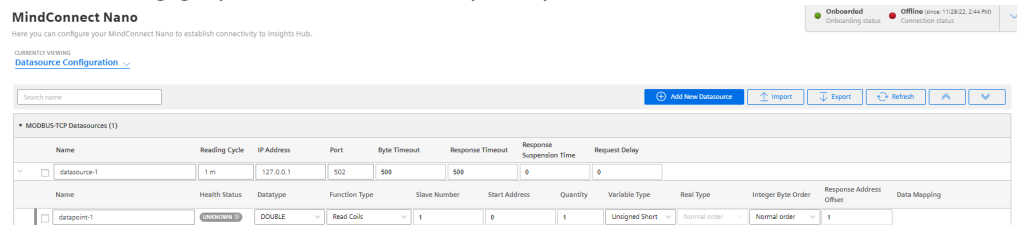
The following parameters are optional and can be adjusted for collecting data from slow Modbus devices:

Parameter	Description
byteTimeout	You can set the timeout interval between two consecutive bytes of the same message received from the Modbus device. If the delay between is longer than the given timeout, an error will be generated. Range: 500ms (default) ... 2000ms
responseTimeout	You can set the timeout interval used to wait for a response from the Modbus device. If the waiting before receiving the response is longer than the given timeout, an error will be generated. Range: 500ms (default) ... 60000ms
responseSuspensionTime	If a byte timeout or a response timeout was detected, then this waiting time is added before the next request is started. During this time any response from the Modbus device will be flushed. Range: 0ms ... 45000ms. (default: 1000ms)
requestDelay	You can set the time between the response from the Modbus device and the next request to this Modbus device. Range: 0ms (default) ... 10000ms

Modbus TCP data point parameter

To collect the data of the Modbus TCP device, you must add new data points.

The following graphic shows the data point parameter window:



The following table shows the parameter of the Modbus TCP data point window:

Parameter	Description
Data Type	Data type supported by backend (e.g. INT).
Function Type	Function type - depends on the type of the data (see manual of the Modbus device). 1: Read Coils, 2: Read Inputs, 3: Read Holding Registers, 4: Read Input Registers . Currently function codes 1..4 are supported.
Slave Number	Address/Slave number of the Modbus device.

Parameter	Description
Start Address	Start address (offset) of the data.
Quantity	Quantity of data to be read.
VariableType	<p>You can choose the following variable types of the Modbus data to be read:</p> <ul style="list-style-type: none"> 1: char : 8 bits, 2: unsigned char : 8 bits, 3: short : 16 bits, 4: unsigned short : 16 bits, 5: long : 32 bits, 6: unsigned long : 32 bits, 7: float : 32 bits, 8: double : 64 bits, 9: ascii-text : 8 bits / char
Real Type	<p>Real type - determines, if the registers of real values have to be processed in reversed order.</p> <ul style="list-style-type: none"> 0: Normal order, 1: Reversed order, 2: Byte swap order, 3: Word swap order, <p>Real types 2 and 3 are available from V03.04.02.01.</p>
Integer Byte Order	<p>It determines in which byte order format the read value will be processed (default: 0)</p> <ul style="list-style-type: none"> 0: Normal order (big-endian) 1: Reversed order (little-endian byte swap) 2: Byte swap order (big-endian byte swap) 3: Word swap order (little-endian) <p>Note: Integer byte order is available for 16 bits and 32 bits variable types only, and it is supported for the version starting from MindConnect Nano V03.04.02.13.</p>
Response Address Offset	Offset of the data to return in register units.
Response Quantity	Quantity of the returned data. Only for string data a value greater than 1 is supported.



TCP Gateway slave number

If you are using a TCP Gateway, you must select the appropriate slave number for the configuration of the data points.

Addressing:

The information is stored in the Modbus Slave device in 4 different tables. 2 tables store on/off discrete values (coils) and two store numerical values (registers). Each coil is 1 bit long and

assigned a data address between 0 and 65535. Each register has the size of 1 word(16 bits/ 2 bytes) and has a data address between 0 and 65535.

Coil/ Register numbers	Data addresses	Function type	Table name
1-9999	0 to 65535	1	Read Coils
10001-19999	0 to 65535	2	Read Inputs
30001-39999	0 to 65535	4	Read Input Registers
40001-49999	0 to 65535	3	Read Holding Registers

Coil / register numbers can be considered as location names since they do not appear in the configuration. The Data Addresses (Start Address) are used in the configuration.

Data Types:

Data type	Length	Data range
BOOLEAN	1 bit	True to False
INT	4 bytes	-2147483648 to 2147483647
LONG	8 bytes	-9223372036854775808 to 9223372036854775807
DOUBLE	8 bytes	-9.999999999999999999999999999999E+125 to 9.999999999999999999999999999999E+125
STRING	1 - 255 bytes	

Modbus data types:

Data type	Length	Data type number	Data range
char	8 bits	1	-128 to 127
unsigned char	8 bits	2	0 to 255
short	16 bits	3	-32768 to 32767
unsigned short	16 bits	4	0 to 65535
long	32 bits	5	-2147483648 to 2147483647
unsigned long	32 bits	6	0 to 4294967295

Data type	Length	Data type number	Data range
float	32 bits	7	-3.402823e+38 to -1.175 495e-38 +1.175 495e-38 to +3.402823e+38
double	64 bits	8	-1.7976931348623158e+308 to -2.2250738585072014e-308 +2.2250738585072014e-308 to +1.7976931348623158e+308
ascii-text	8 bits / char	9	

Examples:

Example	Data type	Function type	Start address	Variable type	Response Address Offset	Response Quantity
Reading the first Holding Register, number 40001, has the start address 0.	int	Read Holding Registers	0	unsigned short or short	0	1
VariableType: 4 or 3 (unsigned short or short)Reading the second Input Register, number 30002, has the start address 1.	int	Read Input Registers	1	unsigned short or short	0	1
Reading a float (32 bits) from Input Register 30108 / 30109, has the start address 107.	double	Read Input Registers	107	float (reads two registers)	0	1
Reading a string. This example describes to read four letters from register 40006 and 40007. Each register contains two characters.	string	Read Holding Registers	5	unsigned short	0	2 (registers)
Reading a Boolean from the Holding Register 40011, has the start address 10. If the Holding Register has the value 0 then it is displayed as 0. If one bit(or more) has the value 1 then it will be displayed as 1.	Boolean	Read Holding Registers	10	unsigned short or short	0	1

Using Modbus register addresses directly

Usually, it is necessary to enter Modbus registered address decremented by 1 as "start address", but it may lead to confusion when comparing the data point's "start address" with the registered address from Modbus device's manual. Therefore, enter a negative value in the "response address offset" field to directly use the registered address from the Modbus device's manual as "start address". In case a negative value has been entered as "response address offset", this value will internally be subtracted from the given "start address" number to read the data from

Modbus device. Positive "response address offset" values will be handled to address specific data within the read data if several registers need to be read to acquire the data from Modbus device as data block and the necessary data itself is only a part of that data block.

Supported since:

- MindConnect Nano V03.04.03.03 b001

Samples:

	Using decremented start address	Using address from manual	Getting data from a read data block
Registered address from Modbus device manual	303	303	64001
Start address	302	303	64001 ¹⁾
Quantity	1	1	27
Response address offset	0	-1	11
Response quantity	1	1	16

¹⁾ Sample start offset as documented in SENTRON PAC manual - can be used without decrementing it by 1.

5.5 Overview of Modbus RTU protocol for MindConnect Nano

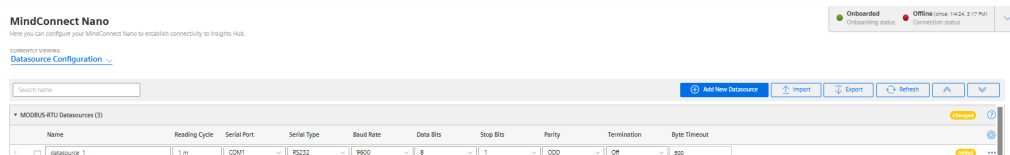
The Modbus protocol is supported for the MindConnect Nano plugin. You can select the protocol type Modbus RTU while adding a new data source in Asset Manager.



Modbus configuration

The Modbus configuration view is only visible after a successful onboarding of the MindConnect Nano. The MindConnect Nano Version must be higher than V3.3.0.2.

The following picture shows the protocol selection in Asset Manager:



For more information about how to create a new data source see chapter [Adding a data source and data points for MindConnect Nano](#).

Modbus RTU data source parameter

In order to connect a Modbus RTU device, you need to select the Modbus RTU protocol and enter the following data source parameters:

Parameter	Description
Serial Port	Available serial ports: <ul style="list-style-type: none">- COM1- COM2
Serial Type	Supported serial communication standards: <ul style="list-style-type: none">- RS232 (default)- RS422- RS485
Baud rate	Baud rate for serial communication: <ul style="list-style-type: none">- 9600 (default)- 2400- 4800- 19200- 38400- 57600- 115200
Data bits	Data bits for serial communication: <ul style="list-style-type: none">- 5- 6- 7- 8 (default)
Stop bits	You can select the number of stop bits for serial communication: <ul style="list-style-type: none">- 1 (default)- 2
Parity	Parity for serial communication: <ul style="list-style-type: none">- Odd- None- Even
Termination	Onboard termination: <ul style="list-style-type: none">- On (default)- Off

The following parameters are optional and can be adjusted for collecting data from slow Modbus devices:

Parameter	Description
-----------	-------------

Parameter	Description
byteTimeout	You can set the timeout interval between two consecutive bytes of the same message received from the Modbus device. If the delay between is longer than the given timeout, an error will be generated. Range: 500ms (default) ... 2000ms
responseTimeout	You can set the timeout interval used to wait for a response from the Modbus device. If the waiting before receiving the response is longer than the given timeout, an error will be generated. Range: 500ms (default) ... 60000ms
responseSuspensionTime	If a byte timeout or a response timeout was detected, then this waiting time is added before the next request is started. During this time any response from the Modbus device will be flushed Range: 0ms ... 45000ms. (default: 1000ms). For Modbus RTU it is recommended to set the value to 10000 ms. If the default value 0 ms is set, any incoming incorrect responses will not be destroyed.
requestDelay	You can set the time between the response from the Modbus device and the next request to this Modbus device. Range: 0ms (default) ... 10000ms

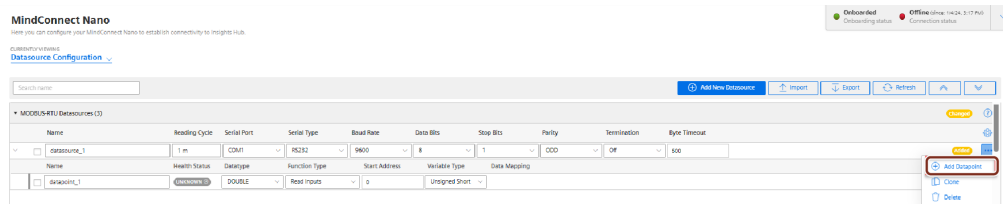
Pin assignment of Modbus protocols

The following table shows the pin assignment for the Modbus RTU protocol:

Pin	RS 485	RS 422	RS 232
1	Data- Transmit / receive data - (I/O) for half-duplex mode	TX- Transmitted data - (O) for full-duplex mode	Not supported
2	Data+ Transmit / receive data+ (I/O) for half-duplex mode	TX+ Transmitted data + (O) for full-duplex mode	RxD Received data (I)
3	-	RX+ Receive data + (I) for full-duplex mode	TxD Transmitted data (O)
4	-	RX- Receive data - (I) for full-duplex mode	Not supported
5	M Signal ground	M Signal ground	M Ground
6	-	-	Not supported
7	-	-	Not supported
8	-	-	Not supported
9	-	-	Not supported

Modbus RTU data point parameter

In order to collect the data of your Modbus RTU device, you must add new data points. The following picture shows the data point parameter window:



The following table shows the parameter of the Modbus RTU data point window:

Parameter	Description
Data Type	Data type supported by backend (e.g. INT).
Function Type	Function type - depends on the type of the data (see manual of the Modbus device). 1: Read Coils 2: Read Inputs 3: Read Holding Registers 4: Read Input Registers Currently function codes 1..4 are supported.
Slave Number	Address/Slave number of the Modbus device.
Start Address	Start address (offset) of the data.
Quantity	Quantity of data to be read.
VariableType	You can choose the following variable types of the Modbus data to be read: 1: char 2: unsigned char 3: short 4: unsigned short 5: long 6: unsigned long 7: float 8: double 9: ASCII-Text
Real Type	Real type - determines, if the registers of real values have to be processed in reversed order 0: Normal order 1: Reversed order 2: Byte swap order 3: Word swap order Real types 2 and 3 are available from V03.04.02.01.

Parameter	Description
Response Address Offset	Offset of the data to return in register units.
Response Quantity	Quantity of the returned data. Only for string data a value greater than 1 is supported.

Addressing:

The information is stored in the Modbus Slave device in 4 different tables. 2 tables store on/off discrete values (coils) and two store numerical values (registers). Each coil is 1 bit long and assigned a data address between 0 and 65535. Each register has the size of 1 word(16 bits/ 2 bytes) and has a data address between 0 and 65535.

Coil/ Register numbers	Data addresses	Function type	Table name
1-9999	0 to 65535	1	Read Coils
10001-19999	0 to 65535	2	Read Inputs
30001-39999	0 to 65535	4	Read Input Registers
40001-49999	0 to 65535	3	Read Holding Registers

Coil / register numbers can be considered as location names since they do not appear in the configuration. The Data Addresses (Start Address) are used in the configuration.

Data Types:

Industrial IoT data types:

Data type	Length	Data range
BOOLEAN	1 bit	True
INT	4 bytes	-2147483648
LONG	8 bytes	-9223372036854775808
DOUBLE	8 bytes	-9.999999999999999999999999999999E+125
STRING	1 - 255 bytes	

Modbus data types:

Data type	Length	Data type number
char	8 bits	1
unsigned char	8 bits	2

Data type	Length	Data type number
short	16 bits	3
unsigned short	16 bits	4
long	32 bits	5
unsigned long	32 bits	6
float	32 bits	7
double	64 bits	8
ascii-text	8 bits / char	9

Examples:

Example	Data type	Function type	Start address	Variable type	Response Address Offset	Response Quantity
Reading the first Holding Register, number 40001, has the Data Address 0.	int	Read Holding Registers	0	unsigned short or short	0	1
VariableType: 4 or 3 (unsigned short or short)Reading the second Input Register, number 30002, has the Data Address 1.	int	Read Input Registers 1	unsigned short or short	0	1	
Reading a float (32 bits) from Input Register 30108 / 30109, has the Data Address 107.	double	Read Input Registers	107	float (reads two registers)	0	1
Reading a string. This example describes to read four letters from register 400006 and 400007. Each register contains two characters.	string	Read Holding Registers	5	unsigned short	0	2 (registers)

Using Modbus register addresses directly

Usually, it is necessary to enter Modbus registered address decremented by 1 as "start address", but it may lead to confusion when comparing the data point's "start address" with the registered address from Modbus device's manual. Therefore, enter a negative value in the "response address offset" field to directly use the registered address from the Modbus device's manual as "start address". In case a negative value has been entered as "response address offset", this value will internally be subtracted from the given "start address" number to read the data from

Modbus device. Positive "response address offset" values will be handled to address specific data within the read data if several registers need to be read to acquire the data from Modbus device as data block and the necessary data itself is only a part of that data block.

Supported since:

- MindConnect Nano V03.04.03.03 b001

Samples:

	Using decremented start address	Using address from manual	Getting data from a read data block
Registered address from Modbus device manual	303	303	64001
Start address	302	303	64001 ¹⁾
Quantity	1	1	27
Response address offset	0	-1	11
Response quantity	1	1	16

¹⁾ Sample start offset as documented in SENTRON PAC manual - can be used without decrementing it by 1.

Technical specification

Protocol	Description	Value	Data rate
Modbus RTU - RS485	Data provider Max cable length	32 1200 m	max. 115,2 kBd
Modbus RTU - RS422	Data provider Max cable length	10 1200 m	max. 115,2 kBd
Modbus RTU - RS232	Data provider Maximum cable length	1 2.5 m 5.0 m 7.5 m 15 m 30 m 60 m 120 m	115,2 kBd 57,6 kBd 38,4 kBd 19,2 kBd 9,6 kBd 4,8 kBd 2,4 kBd

!!!note Maximum cable length depends on the cable type used and maximum length can be increased by using repeaters.

5.6 Overview of SYSTEM protocol for MindConnect Nano

You can use the protocol SYSTEM to implement a system information adapter for analysis purposes. The SYSTEM protocol can collect the following information from MindConnect Nano:

- Overall CPU load
- Overall memory consumption
- Network traffic
- Application specific data
 - CPU load memory consumption
- Disk usage

You can choose the SYSTEM protocol as a data source without configuring specific system parameters, except the reading cycle.

After creating the data source you can configure the following categories in a new data point:

- CPU
- Memory
- Network
- Disk

The following image shows the data point input window for the SYSTEM protocol:



Each category has its own set of variables which will be described in the following.

Application information

Application provides the following available applications:

- MindEdgeRuntimeSystem/java
- DataProvider
- BoxManager
- S7 Plugin
- OPC UA Plugin
- XTools
- Mosquitto

The following table shows the available variables for the applications:

Variable	Data type	Address
Name	string	/application/<subcategory>/name
State	stri	/application/<subcategory>/state
Tgid	int	/application/<subcategory>/tgid
Ngid	int	/application/<subcategory>/ngid
Pid	int	/application/<subcategory>/pid
PPid	int	/application/<subcategory>/ppid
TracerPid	int	/application/<subcategory>/tracerpid
Uid	string	/application/<subcategory>/uid
Gid	string	/application/<subcategory>/gid
FDSIZE	int	/application/<subcategory>/fdsize
Groups	string	/application/<subcategory>/groups
NStgid	int	/application/<subcategory>/nstgid
NSpid	int	/application/<subcategory>/nspid
NSpgid	int	/application/<subcategory>/nspgid
NSsid	int	/application/<subcategory>/nssid
VmPeak	long	/application/<subcategory>/vmpeak
VmSize	long	/application/<subcategory>/vmsize
VmLck	long	/application/<subcategory>/vmlck
VmPin	long	/application/<subcategory>/vmpin
VmHWM	long	/application/<subcategory>/vmhwm
VmRSS	long	/application/<subcategory>/vmrss
VmData	long	/application/<subcategory>/vmdata
VmStk	long	/application/<subcategory>/vmstk
VmExe	long	/application/<subcategory>/vmexe

Variable	Data type	Address
VmLib	long	/application/<subcategory>/vmlib
VmPTE	long	/application/<subcategory>/vmpte
VmPMD	long	/application/<subcategory>/vmpmd
VmSwap	long	/application/<subcategory>/vmswap
HugetlbPages	long	/application/<subcategory>/hugetlbpages
Threads	int	/application/<subcategory>/threads
SigQ	string	/application/<subcategory>/sigq
SigPnd	string	/application/<subcategory>/sigpnd
ShdPnd	string	/application/<subcategory>/shdpnd
SigBlk	string	/application/<subcategory>/sigblk
SigIgn	string	/application/<subcategory>/sigign
SigCgt	string	/application/<subcategory>/sigcgt
CapInh	string	/application/<subcategory>/capinh
CapPrm	string	/application/<subcategory>/capprm
CapEff	string	/application/<subcategory>/capeff
CapBnd	string	/application/<subcategory>/capbnd
CapAmb	string	/application/<subcategory>/capamb
Seccomp	int	/application/<subcategory>/seccomp
Speculation_Store_Bypass	string	/application/<subcategory>/speculation_store_bypass
Cpus_allowed	int	/application/<subcategory>/cpus_allowed
Cpus_allowed_list	string	/application/<subcategory>/cpus_allowed_list
Mems_allowed	int	/application/<subcategory>/mems_allowed
Mems_allowed_list	string	/application/<subcategory>/mems_allowed_list
voluntary_ctxt_switches	int	/application/<subcategory>/voluntary_ctxt_switches

Variable	Data type	Address
nonvoluntary_ctxt_switches	int	/application/<subcategory>/nonvoluntary_ctxt_switches

CPU information

The following table shows the available variables for the category CPU information:

Variable	Data type	Description
LoadAvg1	double	Number of processes in state R(runnable) or D(disk sleep) averaged over 1 minute.
LoadAvg5	double	Number of processes in state R(runnable) or D(disk sleep) averaged over 5 minutes.
LoadAvg10	double	Number of processes in state R(runnable) or D(disk sleep) averaged over 15 minutes.
ProcessCount	string	String containing number of currently runnable processes/threads and number of scheduling processes/threads that currently exist on the system.
LastPidUsed	long	Pid of the process that was most recently created.
CurrentProcessCount	long	Number of currently runnable processes/threads.
ScheduledProcessCount	long	Number of scheduling processes/threads that currently exist on the system.

Disk information

Disk collects information about disk usage and provides the following subcategories:

- persistent_massdata
- persistent_appconfig
- persistent
- root

The following table shows the available variables for the category Disk information:

Variable	Data type	Address	Description
Available	long	/disk/dev/<subcategory>/available	Available space in bytes.

Variable	Data type	Address	Description
Size	long	/disk/dev/<subcategory>/size	Partition size in bytes.
Used	long	/disk/dev/<subcategory>/used	Used space in bytes.

KERNEL information

The following table shows the available variables for the category KERNEL information:

Variable	Data type	Address
OsRelease	string	/kernel/osrelease
OsType	string	/kernel/ostype
Version	string	/kernel/version

Memory information

The following table shows the available variables for the category Memory information:

Variable	Data type	Address	Restrictions
Active	long	/memory/active	
Active(Anon)	long	/memory/active(anon)	
Active	long	/memory/active	
Active(Anon)	long	/memory/active(anon)	
Active(File)	long	/memory/active(file)	
AnonPages	long	/memory/anonpages	
Available	long	/memory/available	
Bounce	long	/memory/bounce	
Buffers	long	/memory/buffers	
Cached	long	/memory/cached	
CommitLimit	long	/memory/commitlimit	
Committed_As	long	/memory/committed_as	
DirectMap4K	long	/memory/directmap4k	

Variable	Data type	Address	Restrictions
DirectMap4M	long	/memory/directmap4m	
Dirty	long	/memory/dirty	
AnonHugePages	long	/memory/anonhugepages	Only supported by MindConnect Nano.
CmaTotal	long	/memory/cmatotal	Only supported by MindConnect Nano.
CmaFree	long	/memory/cmafree	Only supported by MindConnect Nano.
HugePages_Total	long	/memory/hugepages_total	Only supported by MindConnect Nano.
HugePages_Free	long	/memory/hugepages_free	Only supported by MindConnect Nano.
HugePages_Rsvd	long	/memory/hugepages_Rsvd	Only supported by MindConnect Nano.
HugePages_Surp	long	/memory/hugepages_surp	Only supported by MindConnect Nano.
Hugepagesize	long	/memory/hugepagesize	Only supported by MindConnect Nano.
Inactive	long	/memory/inactive	
Inactive(Anon)	long	/memory/inactive(anon)	
Inactive(File)	long	/memory/inactive(file)	
KernelStack	long	/memory/kernelstack	
LowFree	long	/memory/lowfree	
LowTotal	long	/memory/lowtotal	
Mapped	long	/memory/mapped	
MemFree	long	/memory/memfree	
MemTotal	long	/memory/memtotal	
MLocked	long	/memory/mlocked	
NFS_Unstable	long	/memory/nfs_unstable	
PageTables	long	/memory/pagetable	
SHMem	long	/memory/shmem	
SLab	long	/memory/slab	
SReclaimable	long	/memory/sreclaimable	
SUnreclaim	long	/memory/sunreclaim	

Variable	Data type	Address	Restrictions
SwapCached	long	/memory/swapcached	
SwapFree	long	/memory/swapfree	
SwapTotal	long	/memory/swaptotal	
Unevictable	long	/memory/unevictable	
VMAllocChunk	long	/memory/vmallocchunk	
VMAllocTotal	long	/memory/vmalloctotal	
VMAllocUsed	long	/memory/vmallocated	
WriteBack	long	/memory/writeback	
WriteBackTmp	long	/memory/writebacktmp	

Network information

The following table shows the available variables for the category Network information:

Variable	Data type	Address	Description
ArpFilter	long	/network/arpfilter	
BusyPollRxPackets	long	/network/busypollrxpackets	
BytesReceived	long	/network/bytesreceived	Alias for InOctets.
BytesSend	long	/network/bytessend	Alias for OutOctets.
DelayedAckLocked	long	/network/delayedacklocked	Number of delayed acks further delayed because of locked socket.
DelayedAckLost	long	/network/delayedacklost	Number of times that quick ack mode was activated.
DelayedAcks	long	/network/delayedacks	Number of delayed acks sent.
EmbryonicRsTs	long	/network/embryonicrst	Number of resets received for embryonic SYN_RECV sockets.
InBcastOctets	long	/network/inbcastoctets	

Variable	Data type	Address	Description
InBcastPkts	long	/network/inbcastpkts	
IncePkts	long	/network/incepks	
IncSumErrors	long	/network/incsumerrors	
Inect0Pkts	long	/network/inect0pkts	
Inect1Pkts	long	/network/inect1pkts	
InMcastOctets	long	/network/inmcastoctets	
InMcastPkts	long	/network/inmcastpkts	
InNoEctPkts	long	/network/innoectpkts	
InNoRoutes	long	/network/innoroutes	
InOctets	long	/network/inoctets	Bytes received.
InTruncatedPkts	long	/network/intruncatedpkts	Number of truncated packets received.
IpReversePathFilter	long	/network/ipreversepathfilter	
ListenDrops	long	/network/listendrops	Number of SYN's to LISTEN sockets that were dropped.
ListenOverflows	long	/network/listenoverflows	Number of times the listen queue of a socket overflowed
LockDroppedIcmps	long	/network/lockdroppedicmps	ICMP packets dropped because socket was locked.
OfoPruned	long	/network/ofopruned	Packets dropped from out-of-order queue because of socket buffer overrun.
OutBcastOctets	long	/network/outbcastoctets	
OutBcastPkts	long	/network/outbcastpkts	
OutMcastOctets	long	/network/outmcastoctets	
OutMcastPkts	long	/network/outmcastokts	
OutOctets	long	/network/outoctets	Bytes sent.

Variable	Data type	Address	Description
OutOfWindowIcmps	long	/network/outofwindowicmps	ICMP packets dropped because they were out-of-window.
PAWSActive	long	/network/pawsactive	Active connections rejected because of timestamp.
PAWSEstab	long	/network/pawsestab	Packets rejected in established connections because of timestamp.
PAWSPassive	long	/network/pawspassive	Passive connections rejected because of timestamp.
PruneCalled	long	/network/prunecalled	Packets pruned from receive queue because of socket buffer overrun.
RcvPruned	long	/network/rcvpruned	Packets pruned from receive queue.
SyncookiesFailed	long	/network/syncookiesFailed	Number of invalid SYN cookies received.
SyncookiesRecv	long	/network/syncookiesrecv	Number of SYN cookies received.
SyncookiesSent	long	/network/syncookiesent	Number of SYN cookies sent.
TCPAbortFailed	long	/network/tcpabortfailed	Number of times unable to send RST due to no memory.
TCPAbortOnClose	long	/network/tcpabortonclose	Connections reset due to early user close.
TCPAbortOnData	long	/network/tcpabortondata	Connections reset due to unexpected data.
TCPAbortOnLinger	long	/network/tcpabortonlinger	Connections aborted after user close in linger timeout.
TCPAbortOnMemory	long	/network/tcpabortonmemory	Connections aborted due to memory pressure.

Variable	Data type	Address	Description
TCPAbortOnSyn	long	/network/tcpabortonsync	Connections reset due to unexpected SYN.
TCPAbortOnTimeout	long	/network/tcpabortontimeout	Connections aborted due to timeout.
TCPAckSkippedChallenge	long	/network/tcpackskippedchallenge	
TCPAckSkippedFinWait2	long	/network/tcpackskippedfinwait2	
TCPAckSkippedPAWS	long	/network/tcpackskippedpaws	
TCPAckSkippedSEQ	long	/network/tcpackskippedseq	
TCPAckSkippedSYNRecv	long	/network/tcpackskippedsynrecv	
TCPAckSkippedTimeWait	long	/network/tcpackskippedtimewait	
TCPAutoCorking	long	/network/tcpautocorking	
TCPBacklogDrop	long	/network/tcpbacklogdrop	
TCPChallengeAck	long	/network/tcpchallengeack	
TCPDeferAcceptDrop	long	/network/tcpdeferacceptdrop	
TCPDirectCopyFromBacklog	long	/network/tcpdirectcopyfrombacklog	Bytes directly in process context from backlog.
TCPDirectCopyFromPrequeue	long	/network/tcpdirectcopyfromprequeue	Bytes directly received in process context from prequeue.
TCPDSACKIGNoRedNoUndo	long	/network/tcpdsackignorednoundo	
TCPDSACKIGNoRedOld	long	/network/tcpdsackignoredold	
TCPDSACKOfoRecv	long	/network/tcpdsackoforecv	DSACKs for out of order packets
TCPDSACKOfoSent	long	/network/tcpdsackofosent	DSACKs sent for out of order packets.
TCPDSACKOldSent	long	/network/tcpdsackoldsent	DSACKs sent for old packets.
TCPDSACKRecv	long	/network/tcpdsackrecv	DSACKs received.

Variable	Data type	Address	Description
TCPDSACKUndo	long	/network/tcpdsackundo	Congestion window recovered without slow start using DSACK.
TCPFACKReorder	long	/network/tcpfackreorder	Number of times reordering was detected using FACK.
TCPFastOpenActive	long	/network/tcpfastopenactive	
TCPFastOpenActiveFailL	long	/network/tcpfastopenactivefail	
TCPFastOpenCookieReqD	long	/network/tcpfastopencookiereqd	
TCPFastOpenListenOverflow	long	/network/tcpfastopenlistenoverflow	
TCPFastOpenPassive	long	/network/tcpfastopenpassive	
TCPFastOpenPassiveFail	long	/network/tcpfastopenpassivefail	
TCPFastRetrans	long	/network/tcpfastretrans	Fast retransmits.
TCPForwardRetrans	long	/network/tcpforwardretrans	Forward retransmits.
TCPFromZeroWindowAdv	long	/network/tcpfromzerowindowadv	
TCPFullUndo	long	/network/tcpfullundo	Congestion windows fully recovered without slow start.
TCPHPAcks	long	/network/tcphpacks	Predicted acknowledgments.
TCPHPHits	long	/network/tcphphits	Packet headers predicted.
TCPHPHitsToUser	long	/network/tcphphitstouser	Packet headers predicted and directly queued to user.
TCPHYStartDelayCWND	long	/network/tcphystartdelaycwnd	
TCPHYStartDelayDetect	long	/network/tcphystartdelaydetect	
TCPHYStartTrainCWND	long	/network/tcphystarttraincwnd	
TCPHYStartTrainDetect	long	/network/tcphystarttraindetect	
TCPKeepAlive	long	/network/tcpkeepalive	Number of ACK packets sent to keep TCP flow alive.
TCPLossFailures	long	/network/tcplossfailures	Timeouts in loss state.

Variable	Data type	Address	Description
TCPLossProbeRecovery	long	/network/tcplossproberecovery	
TCPLossProbes	long	/network/tcplossprobes	
TCPLossUndo	long	/network/tcplossundo	Congestion windows recovered without slow start after partial ack.
TCPLostRetransmits	long	/network/tcplostretransmits	Retransmits lost.
TCPMD5NotFound	long	/network/tcpmd5notfound	
TCPMD5Unexpected	long	/network/tcpmd5unexpected	
TCPMemoryPressures	long	/network/tcpmemorypressures	Number of times TCP ran low on memory.
TCPMINTTLDrop	long	/network/tcpminttldrop	
TCPMTUPFail	long	/network/tcpmtupfail	
TCPMTUPSuccess	long	/network/tcpmtupsuccess	
TCPOfoDrop	long	/network/tcpofodrop	
TCPOfoMerge	long	/network/tcpofomerge	
TCPOfoQueue	long	/network/tcpofoqueue	
TCPOrigDataSent	long	/network/tcporigdatasent	
TCPPartialUndo	long	/network/tcppartialundo	Congestion windows partially recovered using Hoe heuristic.
TCPPrequeued	long	/network/tcpprequeued	Packets directly queued to recvmmsg prequeue.
TCPPrequeueDropped	long	/network/tcpprequeuedropped	Packets dropped from prequeue.
TCPPureAcks	long	/network/tcppureacks	Acknowledgments not containing data payload received.
TCPRcvCoalesce	long	/network/tcprcvcoalesce	

Variable	Data type	Address	Description
TCPRcvCollapsed	long	/network/tcprcvcollapsed	Packets collapsed in receive queue due to low socket buffer.
TCPRenoFailures	long	/network/tcprenofailures	Timeouts after reno fast retransmit.
TCPRenoRecovery	long	/network/tcprenorecovery	Times recovered from packet loss due to fast retransmit.
TCPRenoRecoveryFail	long	/network/tcprenorecoveryfail	Classic Reno fast retransmits failed.
TCPRenoReorder	long	/network/tcprenoreorder	
TCPreqQFullDoCookies	long	/network/tcpreqqfulldocookies	
TCPreqQFullDrop	long	/network/tcpreqqfulldrop	
TCPretransFail	long	/network/tcpretransfail	
TCPSACKDiscard	long	/network/tcpsackdiscard	
TCPSACKFailures	long	/network/tcpsackfailures	Timeouts after SACK recovery.
TCPSACKMerged	long	/network/tcpsackmerged	
TCPSACKRecovery	long	/network/tcpsackrecovery	Times recovered from packet loss by selective acknowledgements.
TCPSACKRecoveryFail	long	/network/tcpsackRecoveryFails	SACK retransmits failed.
TCPSACKReneging	long	/network/tcpsackreneging	Bad SACK blocks received.
TCPSACKReorder	long	/network/tcpsackreorder	Number of times reordering was detected using SACK.
TCPSACKShifted	long	/network/tcpsackshifted	
TCPSACKShiftFallback	long	/network/tcpsackshiftfallback	
TCPSchedulerFailed	long	/network/tcpschedulerfailed	Times receiver scheduled too late for direct processing.
TCPSlowStartRetrans	long	/network/tcpslowstartretrans	Retransmits in slow start.

Variable	Data type	Address	Description
TCPSpuriousRTOS	long	/network/tcpspuriousrtos	
TCPSpuriousRTXHostQueues	long	/network/tcpspuriousrtxhostqueues	
TCPSYNChallenge	long	/network/tcpsynchallenge	
TCPSYNRetrans	long	/network/tcpsynretrans	
TCPTimeouts	long	/network/tcptimeouts	Other TCP timeouts.
TCPTimeWaitOverflow	long	/network/tcptimewaitoverflow	
TCPToZeroWindowAdv	long	/network/tcptozerowindowadv	
TCPTSReorder	long	/network/tcptsreorder	Number of times reordering was detected using timestamp.
TCPWantZeroWindowAdv	long	/network/tcpwantzerowindowadv	
TCPWinProbe	long	/network/tcpwinprobe	Number of ACK packets to be sent at regular intervals to make sure a reverse ACK packet opening back a window has not been lost.
TW	long	/network/tw	
TWKilled	long	/network/twkilled	TCP sockets finished time wait in slow timer.
TWRecycled	long	/network/twrecycled	Time wait sockets recycled by timestamp.

5.7 Overview of Rockwell protocol for MindConnect Nano

The following image shows the data source input window for the Rockwell protocol:



Rockwell data source parameter

In order to connect a Rockwell device, you need to select the Rockwell protocol and enter the following data source parameters:

Parameter	Description
CPU type	Shows available CPU types: - COMPACT_LOGIX - CONTROL_LOGIX - MICRO_800 - OTHER_CPU Currently not available: - FLEX_LOGIX - MICRO_LOGIX - SOFT_LOGIX - PLC5 - SLC500 Note: The CONTROL_LOGIX and MICRO_800 CPU types are supported from the version MindConnect Nano V03.04.02.06 b002.
IP address	IP address of the Rockwell controller.
Route parameters	Here you can define the routing path parameters - like rack, slot, node, and/or ControlNet port numbers - needed to access the controller (default: 1,0).

Rockwell data point parameter

In order to collect the data of your Rockwell device, you must add new data points.



The following table shows the data point parameter:

Parameter	Description
Data Type	Data type supported by backend (e.g. INT).
Tag	Name of the tag that has to be recorded.

The following table shows the recommended data type selection:

Rockwell data type	Length	Data range ¹	Industrial IoT Data type
BOOL	1 bit	0 or 1	BOOLEAN
SINT	1 byte	-128...+127	INT

Rockwell data type	Length	Data range ¹	Industrial IoT Data type
USINT ²	1 byte	0...255	INT
INT	2 bytes	-32,768...+32,767	INT
UINT ²	2 bytes	0...65,535	INT
DINT	4 bytes	-2,147,483,648...+2,147,483,647	INT
UDINT ²	4 bytes	0...4,294,967,295	INT
REAL	4 bytes	- -3.40282347E38...-1.17549435E-38 (negative values) - 0 - -1.17549435E-38-...3.40282347E38 (positive values)	DOUBLE
SSTRING ²	0 .. 255 bytes	ASCII	STRING
STRING ²	0 .. 255 bytes	ASCII	STRING

¹" Values have been taken from Logix 5000 Controllers I/O and Tag Data Programming Manual.

²" Available since MindConnect Nano/IoT2040 V03.04.04.03 b001, MCSA V03.06.00.17 b003, D-MCSA V03.06.02.02 b002 and MindConnect IoT2050 V03.07.00.03 b003.



PCL data types

Currently only these PLC data types are supported.

Managing MindConnect Nano

6

6.1 Manual firmware update of MindConnect Nano

To prepare your MindConnect Nano for V3 please follow the next two steps:

- Set MindConnect to factory settings
- Load new firmware on your MindConnect Nano

Set MindConnect to factory settings



This procedure is only necessary for boxes that were already connected to V2 (SAP). V1 cannot be upgraded.

1. Set the MindConnect back to factory settings.
2. Create a "ConBox_Commands.json" file with your favorite code editor.

Content of "ConBox_Commands.json":

```
{
  "Commands": [
    {
      "Cmd": "DoFactoryReset",
      "DeviceID": "MindConnectID"
    }
  ]
}
```



MindConnectID has to be replaced with the serial number of the device.

3. Copy the extracted/unzipped fw.tar file to a USB stick (fat32 formatted).
4. Plug the USB stick in the MindConnect hardware.
 - The MindConnect will be restarted automatically.

- The network and agent settings will be reset to their initial state on MindConnect Nano.
- All agent and script logging files will be deleted.

Load new firmware on your MindConnect Nano

1. Download the firmware ZIP file from Siemens Industry Online Support Page.
 - [Firmware download MindConnect Nano](#)
2. Unzip the firmware on your local machine.
3. Copy the unzipped files to a USB stick.
4. Insert the USB stick into the MindConnect Nano and wait until the firmware update is complete.
 - Update carried out: ERROR LED flashes orange
 - MindConnect Nano reboots
 - MAINT LED flashes orange for a few seconds
 - RUN LED orange (not connected to Industrial IoT)
5. Disconnect the USB stick.

6.2 Firmware update in Asset Manager for MindConnect Nano

In Asset Manager, you can update the firmware of your MindConnect Nano.



Outdated version


Please keep your firmware always updated.



Data loss

Please note that data loss can occur during the time of the firmware update. The amount of data loss depends on the reading and uploading cycle.

To perform a firmware update, follow these steps:

1. Select the asset in the "Asset" tab.
2. Open the MindConnect plugin.
3. Click next to "Firmware" on the button .
4. Accept the terms and conditions.

5. To select a firmware release click "Choose another firmware release".
 - The "select firmware" dialog appears.
6. Select product and available version and click "Accept".
7. To update your firmware click "Update your firmware".



Incomplete firmware update

If the connection is interrupted while the new firmware is being downloaded, note that the downloaded content might be lost if the connection cannot be recovered via the box. For example, if the connection is interrupted while 50% of the firmware are downloaded, the process might be canceled. In that case you have to click firmware update on the user interface to restart the process. It is still possible to do manual firmware updates via USB stick.

Canceling an Online Firmware update

If the firmware update process is triggered from UI then the cancel button will be activated. By clicking "Cancel Firmware Update" the ongoing operation will be interrupted and cancelled.



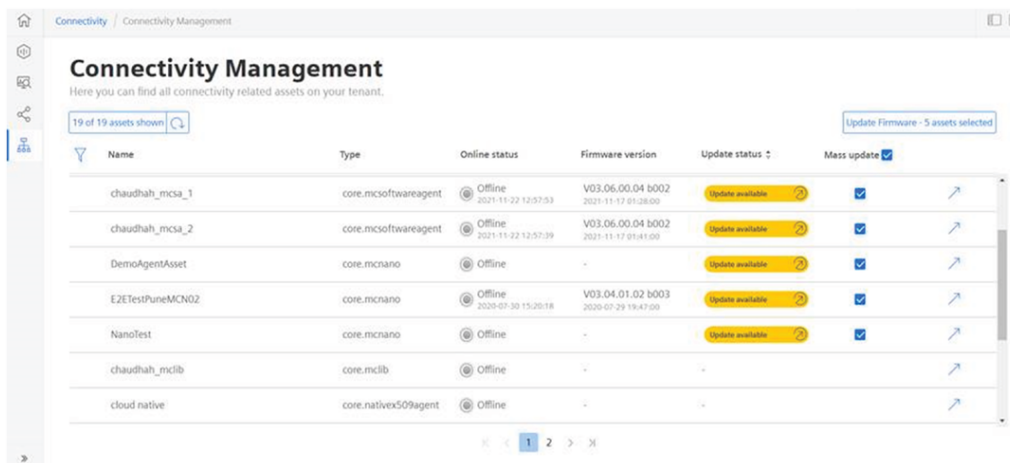
The firmware update operation can take longer time depending on the network quality.

Once the cancellation operation successful, it is possible to re-trigger the online firmware update by clicking "Update Firmware".

Firmware updates for multiple devices

In Asset Manager under "Connectivity Management" tab, you can trigger firmware updates for multiple devices using "Mass update" selection. This allows you to update the selected firmware devices like, MindConnect Nano.

To update the firmware for multiple devices, select assets in the "Connectivity Management" tab under "Mass Update" and click "Update Firmware".



Name	Type	Online status	Firmware version	Update status	Mass update
chaudhah_mcsa_1	core.mcssoftwareagent	Offline 2021-11-22 12:57:53	V03.06.00.04 b002 2021-11-17 01:28:00	Update available	<input checked="" type="checkbox"/>
chaudhah_mcsa_2	core.mcssoftwareagent	Offline 2021-11-22 12:57:39	V03.06.00.04 b002 2021-11-17 01:41:00	Update available	<input checked="" type="checkbox"/>
DemoAgentAsset	core.mcnano	Offline	-	Update available	<input checked="" type="checkbox"/>
EZTestPuneMCN02	core.mcnano	Offline 2020-07-30 15:20:18	V03.04.01.02 b003 2020-07-29 19:47:00	Update available	<input checked="" type="checkbox"/>
NanoTest	core.mcnano	Offline	-	Update available	<input checked="" type="checkbox"/>
chaudhah_mclib	core.mclib	Offline	-	-	<input type="checkbox"/>
cloud native	core.native509agent	Offline	-	-	<input type="checkbox"/>


6.3 Automatic firmware update of MindConnect Nano

In Asset Manager, the firmware of your MindConnect Nano is updated by both manually and automatically. If the auto firmware update option is enabled, MindConnect Nano is automatically updated with the new version when it is released.



- If the system is already updated with a new version available for the device and the auto firmware update option is enabled later, MindConnect Nano will be not triggered to update the previous released version. Auto firmware update will work only for further releases.
- If the auto firmware update is enabled, manual firmware update operation is not allowed.
- If the firmware update job is in progress triggered by the auto firmware update, then "Cancel" button will be activated. By clicking "Cancel" button, the ongoing operation will be interrupted and cancelled. If an auto firmware update trigger is cancelled, it will not be triggered again for the same firmware release.

To perform auto firmware update, follow these steps:

1. Select the asset in the "Asset" tab.
2. Open the MindConnect plugin.
3. Click next to "Firmware" on the button .
4. Click on "Auto FW Update" checkbox in "Firmware Settings" section.
5. Accept the terms and conditions for all releases.

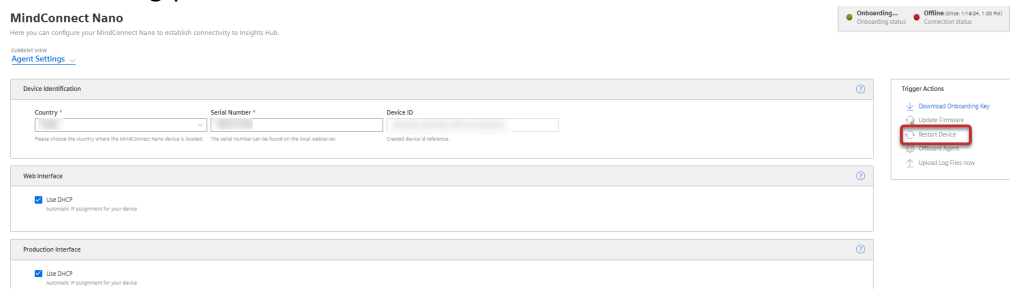
6. Click "Save" button to save the changes.

6.4 Remote restart for MindConnect Nano

By using the remote restart function in the MindConnect Element Plugin, you can restart your device without having to use the on / off button on the device. This makes it easier to restart boxes that are difficult to reach.

If an active restart process is in progress, no further process can be created in the user interface.

The following picture shows the restart device button in the MindConnect Element plugin screen.



6.5 Send commands for MindConnect Nano

The commanding functionality provides you to send data directly from cloud applications and from Insights Hub applications to your devices. The first example of this functionality was remote restart. The send command functionality is extended to cover writing data to the field via S7 and OPC UA plugins. It is possible to write data in the following data types:

- Integer
- Long
- Double
- Boolean
- Float
- String (OPC UA only)



It is important to know the limitations of your devices, before you set a specific value for a datapoint.

- Failure to check value ranges within your controllers may result in damages to your devices. We recommend to set and verify limits within your local program to block values out of device limits.
- If there is an ongoing command for the datapoint, the user will not be allowed to create a new one until the ongoing specific datapoint is finalized.

User interface

The following screen shows the data source configuration screen:

MindConnect Nano
Here you can configure your MindConnect Nano to establish connectivity to Insights Hub.

current view
[Datasource Configuration](#)

Search name Add New Datasource Import

57 Datasources (3)

Name	Reading Cycle	IP Address
<input type="checkbox"/> datasource_1	1 m	127.0.0.1
<input type="checkbox"/> datapoint_1	UNKNOWN	num
<input type="checkbox"/> datapoint_2	UNKNOWN	num
<input type="checkbox"/> datapoint_3	UNKNOWN	num
<input type="checkbox"/> datasource_2	1 m	127.0.0.1
<input type="checkbox"/> datapoint_4	UNKNOWN	bool
<input type="checkbox"/> datapoint_5	UNKNOWN	string
<input type="checkbox"/> datasource_1	1 m	127.0.0.1
<input type="checkbox"/> datapoint_1	UNKNOWN	—

Name	Health Status	Unit	Datatype	Type	Address	On Data Change	Data Mapping
<input type="checkbox"/> datapoint_1	UNKNOWN	num	INT	READ	DB1.DBINT0	true	No mappings available
<input type="checkbox"/> datapoint_2	UNKNOWN	num	LONG	READ&WRITE	DB25.DBReal4	false	No mappings available
<input type="checkbox"/> datapoint_3	UNKNOWN	num	DOUBLE	WRITE	DB1.DBREAL4	false	No mappings available
<input type="checkbox"/> datapoint_4	UNKNOWN	bool	BOOLEAN	READ	DB2.DB83	false	AdvancedMassImportDataOwner / AdvancedMassImportAspect / lec518
<input type="checkbox"/> datapoint_5	UNKNOWN	string	STRING	READ	DB2.DBSTRING3	false	No mappings available
<input type="checkbox"/> datapoint_1	UNKNOWN	—	BOOLEAN	READ	DB15.DBX6.3	false	No mappings available

The "Type" column represents "Data Acquisition Mode" type and provides the following values:

Value	Description
Read	Within this type you can only collect data.
Write	Within this type you can only send data.
Read&Write	Within this type you can collect and send data.

The "Send a value" button is only visible on writeable data point types. This creates a command instance on cloud platform on Edge area for execution.

The "Log" column shows the general result of the latest operation made on the data point itself.

Commanding Device via API

In order to command the device, there are 3 critical id's that needs to be get from the system. Those id's are deviceId, datasourceId and datapointId. To get the id's, follow these steps:

Getting deviceId

You can "GET" the "id" of your device using the "id" of your asset with a below request.
 /api/devicemanagement/v3/devices?assetId={id} In the response, device id will be displayed. For more information, [Device Management Service](#) see documentation.

Getting dataSourceId and datapointId

In the second step, we need to make a request to mindconnectdevicemanagment service to "GET" the id of the datasource and the datapoint.

/api/mindconnectdevicemanagement/v3/devices/{assetId}/dataConfig

In the response of this request the id's will be displayed and should be exdtraced for the final api call. Below is the response for the above request. The "id" field in dataSource item represents the dataSourceId and the dataPointId represents the "id" of the datapoint.

```
{
  "id": "392fe515-5b61-439d-814d-b37b9fa7ef7b",
  "agentId": "25137086d6c141b682d9fcb135c8168e",
  "uploadCycle": "10",
  "configurationId": "1650623649731",
  "dataSources": [
    {
      "id": "326340f5-88c6-409a-a5d9-03e95ce7af94",
      "name": "WindTurbine",
      "description": "datasource_description",
      "protocol": "S7",
      "readCycleInSeconds": "60",
      "protocolData": {
        "ipAddress": "192.168.1.1",
        "manualRackAndSlot": "Automatic",
        "rackNumber": "0",
        "slotNumber": "2"
      },
      "dataPoints": [
        {
          "id": "6efc38d1-b4e7-4e6e-98ce-cb6b3b3f36b2",
          "dataPointId": "36cd56f91c154",
          "name": "Energy",
          "description": "description",
          "unit": "kw",
          "dataType": "DOUBLE",
          "dataPointData": {
            "address": "DB1.DB1",
            "hysteresis": "0",
            "onDataChanged": false,
            "acquisitionType": "READ"
          }
        }
      ]
    }
  ]
}
```

```
    }
  }
]
}
]
```

The final api call is to deploymentworkflow service. For more information, see [Deployment Workflow Service](#). A "POST" request to the below endpoint should be made, /api/deploymentworkflow/v3/instances

The body of the request should be, as below example:

```
{
  "deviceId": "f5168f95-2cc6-4a2c-b598-bfd5dadfc4df",
  "model": {
    "key": "mdsp-core-commandDispatcher"
  },
  "data": {
    "type": "datapoint-write",
    "createdBy": "user@siemens.com",
    "version": "v1.0",
    "to": "some_protocol",
    "payload": {
      "desiredValue": "7",
      "dataSourceId": "10d9c36c-69f7-4c33-8624-f67537e142a4",
      "dataPointId": "b932b9da42f04",
      "protocol": "some_protocol"
    }
  }
}
```

In the below list, there are some fields which require certain data only:

Field Name	Allowed Values
to	OPCUA, S7
protocol	OPCUA, S7
version	v1.0
type	datapoint-write
key	mdsp-core-commandDispatcher

The following list shows the details of the commanding functionality:

- If command operation fails, then the box will set the state of the job to "FAILED" indicating some more details in the message column of the command itself. On the other hand, the state of the job will be "EXECUTED", if the command operation succeeded. These updated states can be checked via Deployment Workflow's endpoint GET /instances too.
- The commanding operations are executed in a chronological order, starting first from the oldest to the newest. The execution of those commands is synchronized with the configuration updates to avoid inconsistencies between the device side content and cloud side content.
 - In order to keep the device configuration content up to date with the backend side do not forget to click the "Apply changes" button. Otherwise, some of your commands will get rejected by the device side, for example a command is sent for a "READ" typed datapoint.
- Boolean type is only working strictly for 'true' and 'false' values.

6.6 USB commands

You can run diagnostic and maintenance USB commands on your MindConnect Nano. The following lists various USB commands and the respective JSON format:

Prerequisites

- A MindConnect device e.g. MindConnect Nano
- A fat32 formatted USB stick

Procedure

To use a USB command, proceed as follows:

1. Make sure your USB stick is fat32 formatted.
2. Create a file named "ConBox_Commands.json" directly on the USB stick.
3. Open the "ConBox_Commands.json" file in a text editor.
4. Enter one of the "JSON Code" commands that you want to execute.
5. Save the changes to the "ConBox_Commands.json".
6. Plug the USB STick into the MindConnect device.

The following list shows various USB commands:

Copy extended logs to USB stick

The following command will collect all agent logs & system files, put them into `ConBox_<UID>_Logs.tar.gz` and `ConBox_<UID>_SystemFiles.tar.gz` files and copy them to the USB stick.

JSON Code:

```
{
  "Commands": [
    {
      "Cmd": "CopySystemFilesToUsbStick",
      "TypeOfFiles": "AllSystemFiles"
    },
    {
      "Cmd": "CopyAgentLogsToUsbStick"
    }
  ]
}
```



For the MindConnect IoT2040 it can take up to 15 minutes to collect all files.

Copy archived logs to USB stick

Some Logs arrive every day at approx. 0 UTC. The extended logs therefore only contain the data of the current day. To get those logs for past days you need to run this command.

The following command will copy all archived log files to a sub directory on the USB stick and delete the copied log files from the archive folder optionally. The USB stick should be larger than 200 MB. Sub directories on the USB stick will have following structure: `ConBox_<BOX ID>/<CurrentTime>Sample:ConBox_J47110815/20180318T191647`

JSON Code:

```
{
  "Commands": [
    {
      "Cmd": "CopyArchivedLogsToUsbStick",
      "DeleteAfterCopy": "false"
    }
  ]
}
```



To store the file locally, disable the log file upload.

Change system time

For the case that the system time should have been changed so that agent runtime is no longer able to connect to Industrial IoT, you can retrieve the backend time to correct the system time. The following command provides the possibility to change the system time manually via ConBox_Commands.json file on the USB stick.

JSON Code:

```
{
  "Commands": [
    {
      "Cmd": "SetSystemTime",
      "NewTime": "2017-09-19T12:34:56"
    }
  ]
}
```



You have to set to the current UTC (Coordinated Universal Time) time.

Change log file upload configuration

You can activate or deactivate the upload of log files via the ConBox_Commands.json file with the ChangeLogFileUploadConfiguration command on a USB stick. This reduces the amount of data that is stored for a environment on the backend side and reduces the mobile data costs for uploading data to Industrial IoT.

If log file upload is disabled, the log files are saved locally. This local space is limited (200 MB). The files are written and after reaching this limit the oldest files are overwritten.

After the log file upload has been reactivated, previously saved MERS log files are uploaded to Industrial IoT (max. 200 MB).

JSON Code:

```
{
  "Commands": [
    {
      "Cmd": "ChangeLogFileUploadConfiguration",
      "EnableUpload": "true"
    }
  ]
}
```

```
]
}
```

You can set the following command parameters:

Parameter	Description
EnableUpload	true: enable log file upload false: disable log file upload



Changing log upload by USB command will not be synchronized with the asset's device configuration on Industrial IoT side.

Perform factory reset

For factory reset a ConBox_Commands . json file can be provided by using the USB stick. The network and BoxMgr settings will be reset to their initial state on MindConnect Nano. All BoxMgr and MERS logging files and temporary files will be then deleted. At the end the MindConnect device will be restarted automatically.



Offboard the asset and device before factory reset execution and perform the factory reset. After the execution, the asset and device should be onboarded again.

The factory reset command will be available on MindConnect Nano version 03.01.00.00 b00x or later.

JSON Code:

```
{
  "Commands": [
    {
      "Cmd": "DoFactoryReset",
      "DeviceID": "_MindConnect_ID_"
    }
  ]
}
```



The Unique ID has to be replaced with the serial number of the device

You can find more information on how to set a MindConnect Element to factory settings in chapter [Manual firmware update of MindConnect Nano](#) and [Manual firmware update of MindConnect IoT2040](#)

Restricted device configuration

The `RestrictDeviceConfiguration` command allows you to enable or disable online device configuration changes via Insights Hub user interface after the MindConnect device has been successfully onboarded.

If you need to adapt the configuration, you have to create a new onboarding file with new e.g. network parameter and insert this data again via USB.



Make sure that all parameters are valid before exporting the configuration file to prevent unwanted changes that can lead to invalid network settings, for example.



Effects on online device configuration

There is no indication in the asset configuration user interface that the device configuration has been restricted on MindConnect Nano device. If you try to change restricted device settings online, a message is logged in the diagnostic file on the MindConnect Nano device that this setting could not be applied.

JSON Code:


```
{
  "Commands": [
    { "Cmd": "RestrictDeviceConfiguration",
      "networkInterfaces": [
        {
          "name": "WebInterface",
          "DHCP": "true",
          "IPv4": "true",
          "SubnetMask": "true",
          "Gateway": "true",
          "DNS": "true"
        },
        {
          "name": "ProductionInterface",
          "DHCP": "true",
          "IPv4": "true",
          "SubnetMask": "true",
          "Gateway": "true",

```

```
        "DNS": "true"
      }
    ]
  }
}
```

To prevent changes of network interface settings, the following parameters can be specified in the JSON command:

Parameter	Description
name	Name of the network interface. Possible values: - WebInterface - ProductionInterface
DHCP	- false: DHCP setting can be changed (default) - false: DHCP setting can be changed (default)
IPv4	- true: IP address can not be changed - false: IP address can be changed (default)
SubnetMask	- true: Subnet mask can not be changed - false: Subnet mask can be changed (default)
Gateway	- true: Gateway address can not be changed - false: Gateway address can be changed (default)
DNS	- true: DNS address can not be changed - false: DNS address can be changed (default)



- If you only provide some of the above mentioned parameters, then the omitted parameters will be ignored. This means that either the last restricted device configuration of these parameters will be used, or they will be treated as not set.
- To enable device configuration changes of specific parameters again, a JSON command containing those parameters with value "false" can be provided via USB stick.
- To turn it off a new JSON command file needs to be provided with "false" for all parameters.

Check network

Command description To find out, if the MindConnect device is successfully connected to Insights Hub using the current onboarding configuration, a ConBox_Commands.json file containing "CheckNetwork" command on an USB stick can be used. After the network analysis has been performed, the analysis result will be written to the USB stick. Depending on the current network situation, it might be possible that the execution of all internally executed tests will take some minutes.



This command is available from the version MindConnect Nano firmware V03.04.02.10 b003.

CheckNetwork command

Using following JSON command in a ConBox_Commands.json file, a network analysis will be performed and the result will be written to the USB stick:

```
{
  "Commands": [
    {
      "Cmd": "CheckNetwork",
      "NetworkChecks": [
        {
          "CheckCloudConnection": "true"
        }
      ]
    }
  ]
}
```

Parameter	Description
CheckCloudConnection	- true: Cloud connection check will be executed - false: Cloud connection check will not be executed.
NetworkChecks	- List of checks that have to be executed. - Currently CheckCloudConnection check is available only.

6.7 Result of CheckCloudConnection file

After the CheckCloudConnection network check has been executed, a result file will be copied to the USB stick.

The name of the result file has the following format:

- ConBox_<serial number>_<time stamp>_CloudConnectionCheck.log

The result file contains following sections:

- Test output
 - Contains detailed information about the executed test steps according to the current network configuration of the MindConnect device.
- Test summary
 - Contains following information.
 - Start and end time of the test.
 - Current IP addresses of web and production interfaces.
- Result of trying to connect to Insights Hub directly without proxy.
- Result of trying to connect to Insights Hub via proxy server, if a proxy server had been configured.
 - Information about the current proxy configuration.

6.8 Troubleshooting

Error	Problem	Possible cause	Possible remedy
-------	---------	----------------	-----------------

Error	Problem	Possible cause	Possible remedy
MindConnect Nano L1- RUN/STOP	<ul style="list-style-type: none"> - MindConnect Element cannot onboard to Insights Hub - No Internet connection 	<p>Network connection problem:</p> <ul style="list-style-type: none"> - Either IP address is not valid or configured IP address already exists in the network of MindConnect Nano - Either disabled or invalid configured Proxy - Proxy is offline - Interrupted connection - IP address is configured via DHCP, but there is no internet connection. - Firewall blocks MindConnect Elements - Your company's router is damaged 	<ul style="list-style-type: none"> - Check your configuration connection: If configured IP and Proxy are valid - Check physical connection: your company's router, Ethernet cables etc.
MindConnect Nano L2 - ERROR LED	<ul style="list-style-type: none"> - Although an asset uploaded to Insights Hub Monitor. - MindConnect Nano cannot onboard to Insights Hub 	<ul style="list-style-type: none"> - No connection to data source - The address of the data point configuration can be incorrect - Incorrect asset configuration - Not accomplished asset configuration (Network Connection) - Invalid ID of MindConnect Nano 	<p>Check the connection between MindConnect Nano and data sources (Plant Network ports and cables-S7/OPC UA Server)</p> <ul style="list-style-type: none"> - Check the connection between MindConnect Nano and data sources (Plant Network ports and cables-S7/OPC UA server) - Check data sources configuration. - Check configuration of each data point. - Check configuration of your asset (Network Configuration). - Check if MindConnect Nano ID is correct.

Error	Problem	Possible cause	Possible remedy
MindConnect Nano L2- ERROR LED blinking RED	Data loss	MindConnect Nano was offline for too long, so it could not send data to Insights Hub and the storage is full.	<ul style="list-style-type: none"> - Check outbound connection - Check Ethernet cables - Check Proxy (IP address, User Authentication can be required)
MindConnect Nano L2- ERROR LED RED	Firmware update error	<ul style="list-style-type: none"> - Technical problems with a new version of firmware - Incorrect firmware type - Certificate problems, Authentication problems etc. 	<ul style="list-style-type: none"> - Make sure that you are using the correct firmware type (MindConnect Nano). - MindConnect Nano will automatically restart and then firmware update will be carried out after a while. - If MindConnect Nano does not restart automatically, then restart it manually: by switching on and off the "power button" on MindConnect Nano.
MindConnect Nano L3- MAINT LED ORANGE	Onboarding failed	<ul style="list-style-type: none"> - Configuration file on the USB stick is invalid - Internal error 	<ul style="list-style-type: none"> - Make sure that the data on the USB stick still is valid. (The data, which you exported to the USB stick, is still valid only for 7 days.). If it is expired, use the asset configuration to export a new valid configuration to your USB stick. - Check asset configuration and reconfigure, if it is required and export a new configuration to USB stick.

Error	Problem	Possible cause	Possible remedy
MindConnect Nano L3- MAINT LED blinking RED	<ul style="list-style-type: none"> - Problems with USB stick - Although no LED lights are shown (except PC ON/WD with constant GREEN) and internet connection is OK, MindConnect Nano still cannot be onboarded to Insights Hub - System Error 	<ul style="list-style-type: none"> - Incorrect formatting of USB stick (no FAT or FAT32) - USB stick is damaged - Write error, USB stick is write-protected or its memory is full. - Configuration file is invalid (damaged, illegible) - See also below Problems with USB stick: - Configuration file in USB stick cannot be read or accepted by MindConnect Nano: - Problems with configuration file - Configuration file is located in an incorrect directory. - Expired configuration file on the USB stick - The configuration file is invalid or not copied to the USB stick Unknown 	<ul style="list-style-type: none"> Check format and partition of the USB stick. It must be FAT or FAT32 formatted and may only contain a single partition. - Try a new USB stick - Remove and plug an USB stick into MindConnect Nano - Check the diagnosis file on the USB stick. - Check the name of the configuration file. It must have MindConnect Nano ID. Do not change the name of the downloaded file. - Check whether the directory is valid. (Configuration file must be in the root directory of the USB stick.) - Make sure that the data on the USB stick is still valid. (The data, which you exported to the USB stick, is still valid only for 7 days.) If it is expired, use the asset configuration to export a new valid configuration to your USB stick. - Switch MindConnect Nano off and on again. - If the problem persists, contact your Siemens Support/„Expert Center“.

USB commands

You can run diagnostic and maintenance USB commands on your MindConnect Nano.

The following lists various USB commands and the respective JSON format:

Prerequisites

- A MindConnect device e.g. MindConnect Nano

- A fat32 formatted USB stick

Procedure

To use a USB command, proceed as follows:

1. Make sure your USB stick is fat32 formatted.
2. Create a file named "ConBox_Commands.json" directly on the USB stick.
3. Open the "ConBox_Commands.json" file in a text editor.
4. Enter one of the "JSON Code" commands that you want to execute.
5. Save the changes to the "ConBox_Commands.json".
6. Plug the USB STick into the MindConnect device.

The following list shows various USB commands:

Copy extended logs to USB stick

The following command will collect all agent logs & system files, put them into ConBox_<UID>_Logs.tar.gz and ConBox_<UID>_SystemFiles.tar.gz files and copy them to the USB stick.

JSON Code:

```
{
  "Commands": [
    {
      "Cmd": "CopySystemFilesToUsbStick",
      "TypeOfFile": "AllSystemFiles"
    },
    {
      "Cmd": "CopyAgentLogsToUsbStick"
    }
  ]
}
```



For the MindConnect IoT2040 it can take up to 15 minutes to collect all files.

Copy archived logs to USB stick

Some Logs arrive every day at approx. 0 UTC. The extended logs therefore only contain the data of the current day. To get those logs for past days you need to run this command.

The following command will copy all archived log files to a sub directory on the USB stick and delete the copied log files from the archive folder optionally. The USB stick should be larger than

200 MB. Sub directories on the USB stick will have following structure: ConBox_<BOX ID>/<CurrentTime>Sample:ConBox_J47110815/20180318T191647

JSON Code:

```
{
  "Commands": [
    {
      "Cmd": "CopyArchivedLogsToUsbStick",
      "DeleteAfterCopy": "false"
    }
  ]
}
```



To store the file locally, disable the log file upload.

Change system time

For the case that the system time should have been changed so that agent runtime is no longer able to connect to Industrial IoT, you can retrieve the backend time to correct the system time.

The following command provides the possibility to change the system time manually via ConBox_Commands.json file on the USB stick.

JSON Code:

```
{
  "Commands": [
    {
      "Cmd": "SetSystemTime",
      "NewTime": "2017-09-19T12:34:56"
    }
  ]
}
```



You have to set to the current UTC (Coordinated Universal Time) time.

Change log file upload configuration

You can activate or deactivate the upload of log files via the ConBox_Commands.json file with the ChangeLogFileUploadConfiguration command on a USB stick. This reduces the amount of data that is stored for an environment on the backend side and reduces the mobile data costs for uploading data to Industrial IoT.

If log file upload is disabled, the log files are saved locally. This local space is limited (200 MB). The files are written and after reaching this limit the oldest files are overwritten.


After the log file upload has been reactivated, previously saved MERS log files are uploaded to Industrial IoT (max. 200 MB).

JSON Code:

```
{
  "Commands": [
    {
      "Cmd": "ChangeLogFileUploadConfiguration",
      "EnableUpload": "true"
    }
  ]
}
```

You can set the following command parameters:


Parameter	Description
EnableUpload	true: enable log file upload false: disable log file upload



Changing log upload by USB command will not be synchronized with the asset's device configuration on Industrial IoT side.

Perform factory reset

For factory reset a ConBox_Commands . json file can be provided by using the USB stick. The network and BoxMgr settings will be reset to their initial state on MindConnect Nano. All BoxMgr and MERS logging files and temporary files will be then deleted. At the end the MindConnect device will be restarted automatically.



Offboard the asset and device before factory reset execution and perform the factory reset. After the execution, the asset and device should be onboarded again.

The factory reset command will be available on MindConnect Nano version 03.01.00.00 b00x or later.

JSON Code:

```
{
  "Commands": [
    {
      "Cmd": "DoFactoryReset",
    }
  ]
}
```

```

    "DeviceID": "_MindConnect_ID_"
  }
]
}

```



The Unique ID has to be replaced with the serial number of the device

You can find more information on how to set a MindConnect Element to factory settings in chapter [Manual firmware update of MindConnect Nano](#) and [Manual firmware update of MindConnect IoT2040](#)

Restricted device configuration

The RestrictDeviceConfiguration command allows you to enable or disable online device configuration changes via Insights Hub user interface after the MindConnect device has been successfully onboarded.

If you need to adapt the configuration, you have to create a new onboarding file with new e.g. network parameter and insert this data again via USB.



Make sure that all parameters are valid before exporting the configuration file to prevent unwanted changes that can lead to invalid network settings, for example.



Effects on online device configuration

There is no indication in the asset configuration user interface that the device configuration has been restricted on MindConnect Nano device. If you try to change restricted device settings online, a message is logged in the diagnostic file on the MindConnect Nano device that this setting could not be applied.

JSON Code:

```

{
  "Commands": [
    { "Cmd": "RestrictDeviceConfiguration",
      "networkInterfaces": [
        {
          "name": "WebInterface",
          "DHCP": "true",
          "IPv4": "true",
          "SubnetMask": "true",
          "Gateway": "true",

```

```
        "DNS": "true"
      },
      {
        "name": "ProductionInterface",
        "DHCP": "true",
        "IPv4": "true",
        "SubnetMask": "true",
        "Gateway": "true",
        "DNS": "true"
      }
    ]
  }
}
```

To prevent changes of network interface settings, the following parameters can be specified in the JSON command:

Parameter	Description
name	Name of the network interface. Possible values: - WebInterface - ProductionInterface
DHCP	- false: DHCP setting can be changed (default) - false: DHCP setting can be changed (default)
IPv4	- true: IP address can not be changed - false: IP address can be changed (default)
SubnetMask	- true: Subnet mask can not be changed - false: Subnet mask can be changed (default)
Gateway	- true: Gateway address can not be changed - false: Gateway address can be changed (default)
DNS	- true: DNS address can not be changed - false: DNS address can be changed (default)



- If you only provide some of the above mentioned parameters, then the omitted parameters will be ignored. This means that either the last restricted device configuration of these parameters will be used, or they will be treated as not set.
- To enable device configuration changes of specific parameters again, a JSON command containing those parameters with value "false" can be provided via USB stick.
- To turn it off a new JSON command file needs to be provided with "false" for all parameters.

Check network

Command description To find out, if the MindConnect device is successfully connected to Insights Hub using the current onboarding configuration, a ConBox_Commands.json file containing "CheckNetwork" command on an USB stick can be used. After the network analysis has been performed, the analysis result will be written to the USB stick.

Depending on the current network situation, it might be possible that the execution of all internally executed tests will take some minutes.



This command is available from the version MindConnect Nano firmware V03.04.02.10 b003.

CheckNetwork command

Using following JSON command in a ConBox_Commands.json file, a network analysis will be performed and the result will be written to the USB stick:

```
{
  "Commands": [
    {
      "Cmd": "CheckNetwork",
      "NetworkChecks": [
        {
          "CheckCloudConnection": "true"
        }
      ]
    }
  ]
}
```

Parameter	Description
CheckCloudConnection	<ul style="list-style-type: none">- true: Cloud connection check will be executed- false: Cloud connection check will not be executed.
NetworkChecks	<ul style="list-style-type: none">- List of checks that have to be executed.- Currently CheckCloudConnection check is available only.

Result of CheckCloudConnection file

After the CheckCloudConnection network check has been executed, a result file will be copied to the USB stick.

The name of the result file has the following format:

- ConBox_<serial number>_<time stamp>_CloudConnectionCheck.log

The result file contains following sections:

- Test output
 - Contains detailed information about the executed test steps according to the current network configuration of the MindConnect device.
- Test summary
 - Contains following information.
 - Start and end time of the test.
 - Current IP addresses of web and production interfaces.
- Result of trying to connect to Insights Hub directly without proxy.
Result of trying to connect to Insights Hub via proxy server, if a proxy server had been
- configured.
 - Information about the current proxy configuration.

Log files

Overview

Log files are generated continuously and automatically uploaded when the "Auto Log Upload" is enabled in the following scenarios:

- Log files are uploaded every time the device is started.
- All the log files are uploaded after midnight.
- Once a log file reaches its maximum file size, it is uploaded as zip file.
- Using "Upload Logs Now" option to upload log files manually.

Log files are uploaded in tar.gz format with a maximum file size limit of 10MB with following MindConnect Firmware version.

- MindConnect Nano V03.04.04.03 b001.

Concatenating log file archive parts

Linux

On Linux, the downloaded log file archive parts is concatenated as follows:

```
cat fileArchiv.tar.gz.parta* > fileArchiv.tar.gz
```

Windows

On Windows, the downloaded log file archive parts is concatenated as follows:

```
type fileArchiv.tar.gz.parta* > fileArchiv.tar.gz
```

For information on USB commands troubleshooting, refer to [USB commands](#).

7.1 Using quality code

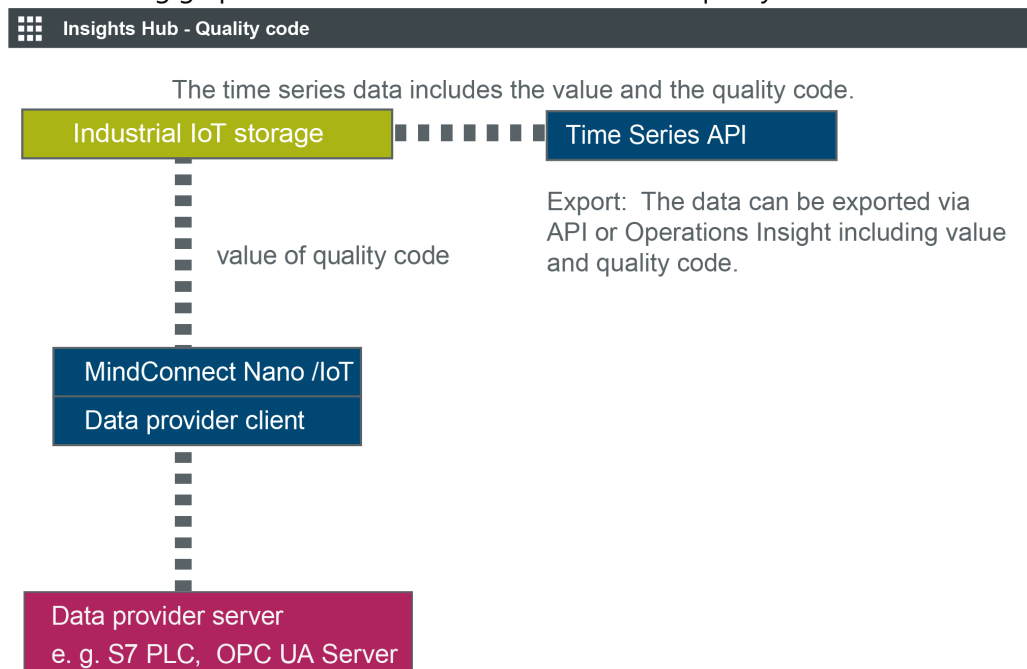
The quality code measures the quality of the connection from a data provider to the MindConnect Nano. In Insights Hub you can enable or disable the transmission of the quality code.

You can switch the quality code on or off:

- **TRUE:** The quality code is transferred with the data to Industrial IoT.
- **FALSE:** The quality code is not transferred with the data to Industrial IoT. Erroneous or bad transmissions and signal losses can not be detected by Insights Hub. Instead the value 0 will be stored for the missing data. You can find more information about the corresponding values in chapter [Quality codes OPC UA and S7 protocol](#).

If you export the time series value (e.g. with Insights Hub Monitor) you can see the data points with its connected quality code. This helps you to analyze time ranges where no data is available. This will avoid drawing false conclusions from a null value.

The following graphic shows a dataflow overview of the quality code:



If the data provider (e.g. S7 PLC, OPC UA Server) is not available, the MindConnect Nano or the data provider client will send up a "0" with a quality code. The value of the quality code shows that this is not a valid data.

- If the function "Quality code" is enabled for this data point all the information can be stored in the database. The database creates a request to Insights Hub Monitor to show the bad quality code information in a chart.
- If the function "Quality code" is disabled, the value 0 will be saved anyway. But you cannot figure out if this data is valid.

7.2 Quality codes OPC UA and S7 protocol

The following table is based on the OPC UA standard and shows the possible quality codes for OPC UA and S7:

Quality code (hex)	Quality code (dec)	Short description	Long description
0x00000000	0	GOOD	Good, all items could be read.
0x00BA0000	12189696	GOOD_INCOMPLETE	Good results may be incomplete - at least one item could be read.
0x002F0000	3080192	GOOD_OVERLOAD	Sampling has slowed down due to resource limitations.
0x40920000	1083310080	UNCERTAIN_INITIAL_VALUE	Uncertain Initial Value - initial value.
0x80000000	-2147483648	BAD	Bad - generic error.
0x80020000	-2147352576	BAD_INTERNAL_ERROR	An internal error occurred as a result of a programming or configuration error.
0x80050000	-2147811328	BAD_COMMUNICATION_ERROR	A low level communication error occurred.
0x80090000	-2146893824	BAD_UNKNOWN_RESPONSE	An unrecognized response was received from the server.
0x800A0000	-2146828288	BAD_TIMEOUT	Bad Timeout - Timeout occurred.
0x800B0000	-2146762752	BAD_SERVICE_UNSUPPORTED	Bad Service Unsupported - service unsupported.
0x80100000	-2146435072	BAD_TOO_MANY_OPERATIONS	Bad Too Many Operations - The request could not be processed because it specified too many operations.

Quality code (hex)	Quality code (dec)	Short description	Long description
0x80130000	-2146238464	BAD_SECURITY_CHECKS_FAILED	Indicates security problems.
0x801F0000	-2145452032	BAD_ACCESS_DENIED	Bad Access Denied - Access denied.
0x80210000	-2145320960	BAD_IDENTITY_TOKEN_REJECTED	Bad Identity Token Rejected.
0x80330000	-2144141312	BAD_NODE_ID_INVALID	The syntax of the node ID is not valid.
0x80340000	-2144075776	BAD_NODE_ID_UNKNOWN	Bad item address.
0x80360000	-2143944704	BAD_INDEX_RANGE_INVALID	The syntax of the index range parameter is invalid.
0x803A0000	-2143682560	BAD_NOT_READABLE	Read operation(s) has problem.
0x803B0000	-2143617024	BAD_NOT_WRITABLE	Write operation(s) has problem.
0x803D0000	-2143485952	BAD_NOT_SUPPORTED	Operation is not supported.
0x803E0000	-2143420416	BAD_NOT_FOUND	A requested item was not found or a search operation ended without success.
0x804F0000	-2142306304	BAD_SERVER_URI_INVALID	The ServerUri is not a valid URI.
0x80740000	-2139881472	BAD_TYPE_MISMATCH	Type conversion has problem.
0x807D0000	-2155675648	BAD_TCPSERVER_TOO_BUSY	The server cannot process the request because it is too busy.
0x80820000	-2138963968	BAD_TCP_INTERNAL_ERROR	TCP internal error occurred.
0x80860000	-2138701824	BAD_SECURE_CHANNEL_CLOSED	The secure channel has been closed.
0x80890000	-2138505216	BAD_CONFIGURATION_ERROR	Lack of configuration info.
0x808D0000	-2138243072	BAD_OUT_OF_SERVICE	OPC service has problems.
0x809B0000	-2137325568	BAD_NO_DATA	No data exists for the requested time range or event filter.
0x809D0000	-2137194496	BAD_DATA_LOST	Data is missing due to collection started/stopped/lost.
0x80AB0000	-2136276992	INVALID_ARGUMENT	Bad Invalid Argument - all items could not be read.
0x80AC0000	-2136211456	BAD_CONNECTION_REJECTED	Connection problem.

Quality code (hex)	Quality code (dec)	Short description	Long description
0x80AD0000	-2136145920	BAD_DISCONNECT	The server has disconnected from the client.
0x80AF0000	-2136014848	BAD_INVALID_STATE	The operation cannot be completed because the object public String getValue() { return value; } is closed.
0x80B80000	-2135425024	BAD_REQUEST_TOO_LARGE	Request too large.
0x80D70000	-2133393408	BAD_BOUND_NOT_FOUND	No data found to provide upper or lower bound value.
0X00008104	33028	BAD_DATABLOCK_ACCESS_DENIED	Read operation cannot be performed because of invalid access to data block

7.3 Quality codes Modbus protocol

The following table shows the possible quality codes for devices that use the Modbus protocol:

Quality code (hex)	Quality code (dec)	Short description	Long description
0x00000000	0	Good	No error occurred
0x80050000	-2147155968	Communication error	Modbus connection reset for connection
0x80310000	-2144272384	No communication	Communication with the data source is defined but not established and there is no last known value available
0x803A0000	-2143682560	Not readable	The access level does not allow reading or subscribing to the node
0x80790000	-2139553792	No subscription	There is no subscription available for this session
0x80AB0000	-2136276992	Invalid argument	One or more arguments are invalid
0x80330000	-2144141312	NodeIdInvalid	Parser bad node syntax
0x809E0000	-2137128960	Data unavailable	Expected data is unavailable for the requested time range due to an unmounted volume, an offline archive or tape, or similar reason for temporary unavailability

Quality code (hex)	Quality code (dec)	Short description	Long description
0x809F0000	-2137063424	Entry exists	The data or event was not successfully updated because no matching entry exists
0x803C0000	-2143551488	Out of range	The value was out of range
0x80270000	-2144927744	Session not activated	The session cannot be used because ActivateSession has not been called
0x808B0000	-2138374144	Device failure	There has been a failure in the device/data source that generates the value that has affected the value
0x803D0000	-2143485952	Not supported	The server cannot retrieve a bound for the variable
0x80630000	-2140995584	Type definition invalid	The type definition node ID does not reference an appropriate type node
0x800D0000	-2146631680	Server not connected	The operation could not complete because the client is not connected to the server.
0x80730000	-2139947008	Write not supported	The server does not support writing the combination of value status and timestamps provided
0x807D0000	-2139291648	Data source busy	Previous read request of data source is still in progress

7.4 Quality codes SYSTEM protocol

The following table shows the possible quality codes for devices that use the SYSTEM adapters:

Quality code (hex)	Quality code (dec)	Short description	Long description
0x00000000	0	Good	No error occurred
0x803A0000	-2151284736	Not readable	The access level does not allow the operation the to the Node (reading or subscribing)
0x807D0000	-2139291648	Data source busy	Previous read request of data source is still in progress

7.5 Quality codes Rockwell protocol

The following table shows the possible quality codes for devices that use the Rockwell protocol:

Quality code (hex)	Quality code (dec)	Short description	Long description
0x00000000	0	Good	No error occurred
0x80010000	-2147418112	Unexpected error	An unexpected error occurred
0x80020000	-2147352576	Internal error	An internal error occurred
0x80030000	-2147287040	Out of memory	Not enough memory for operation
0x80050000	-2147155968	Communication error	Connection to device failed
0x80090000	-2146893824	Unknown response	Received an unsolicited message
0x800A0000	-2146828288	Timeout	The operation timed out
0x800B0000	-2146762752	Service unsupported	Service is not supported
0x800F0000	-2146500608	Nothing to do	Nothing to send
0x801F0000	-2145452032	Access denied	User does not have permission for requested operation
0x80330000	-2144141312	Invalid tag	Requested item (tag) is not known
0x805E0000	-2141323264	Node does not exist	Info file does not exist
0x80630000	-2140995584	Invalid type definition	The type definition does not reference an appropriate type
0x80740000	-2139881472	Type mismatch	Data type is not supported
0x80780000	-2139619328	Too many publish requests	Pending buffer is full
0x807D0000	-2139291648	Data source busy	Previous read request of data source is still in progress
0x80840000	-2138832896	Request interrupted	Request interrupted
0x80850000	-2138767360	Request timeout	Receive timeout occurred
0x80B10000	-2135883776	No data available	No more data available
0x80B60000	-2135556096	Invalid syntax	Syntax error
0x80B80000	-2135425024	Request too large	Buffer is too small
0x80CE0000	-2133983232	Invalid response	Received an invalid reply

7.6 Offline buffer

In order to calculate the buffer time for MindConnect Nano or MindConnect IoT2040 you can use the following formula:

$$\text{Buffer time} = \frac{\text{Buffer size}}{\text{Number of data points (Read cycle)} * \text{Bytes per data point}}$$

MindConnect Nano data buffer is 500 MB. When the buffer is at full capacity, the new incoming data will overwrite the oldest data.

The following example shows the buffer time for a MindConnect Nano with 500 MB data buffer: In this case, 120 bytes is the memory needed for one data point.

$$\text{Buffer time} = \frac{500,000,000 \text{ byte}}{250 \text{ 1/s} * 120 \text{ byte}} = 16,666.6 \text{ s} \approx 4.6 \text{ hrs}$$

7.7 Read performance MindConnect Elements

All MindConnect Elements have a clear limitation on soft limits(unmonitored number of data points and data connections). The only fixed limit is 1050 data points. Within the soft limits, you can configure data connections and data points without reaching the limits. For 250 datapoints read per second, the upload bandwidth should be at least 0.5 MB/s.

The following describes how to proceed if you want to exceed these limits.

Tested MindConnect limits

	MindConnect Nano
Max mount of data points can be configured (Fix)	1050
Data points OPC UA / S7 / Modbus TCP	250
(read per second)	
Connections to data providers	30
String data points	10
Data points - Modbus RTU RS485 / RS422 - 9,6 kBd	20
Data points - Modbus RTU RS485 / RS422 - 19,2 kBd	35
Data points - Modbus RTU RS485 / RS422 - > = 38,4 kBd	80

	MindConnect Nano
Data points - Modbus RTU RS232 - 57,6 kBd	40

Adaptions of the limits

Subsequently listed are the parameters the influence the read performance of a MindConnect Element. By varying one of these parameters all others will be affected as well. These are the variables:

- Performance of MindConnect hardware
- Number of connections to data providers
- Type of connection to providers (S7 / OPC UA / Modbus / ...)
- Amount of data points per second
- Type of data points (String or no string)
- Performance of data provider (PLC, OPC Server)



- Reading data from a data provider will always cause an effect on it. The performance of the data provider (PLC / OPC Server / Modbus) is one of the most important limiting factors. If more devices (e.g. OPC server, HMIs, SCADA systems,...) are connected to this data provider (e.g. PLC) the reading cycle can rise.
- For MindConnect Software Agent, the readings are not applicable as it depends on the hardware for a particular agent and configuration.

Example calculations (MindConnect Nano)

- It can be possible to read 300 data points per second out of a PLC. This is possible if the PLC is powerful and just one data provider is used.
- It can be possible to read 1000 data points out of 30 data providers if the reading cycle will be switched to 4sec.
- It can be possible to read out of 35 data providers if you just read 200 data points per second.



To make sure a configuration works for your setup it needs to be monitored over a longer period of time.

System Test results

The following tables show how long it takes to read a data item from one of the devices. These are not fixed values but an indication of how the devices would react.

MindConnect NANO:

Type	MS per data point read: Min	MS per data point read: Mid	MS per data point read: Max
S7 300	1,13	1,74	3,08
S7 1500	0,37	1,60	3,89
S7 1200	3,50	5,18	7,50
S7 ET200S CPU	0,89	2,09	3,71
OPC UA @ PC ¹⁾	2,38	6,64	8,53
OPC UA@ PLC	2,66	3,27	4,69

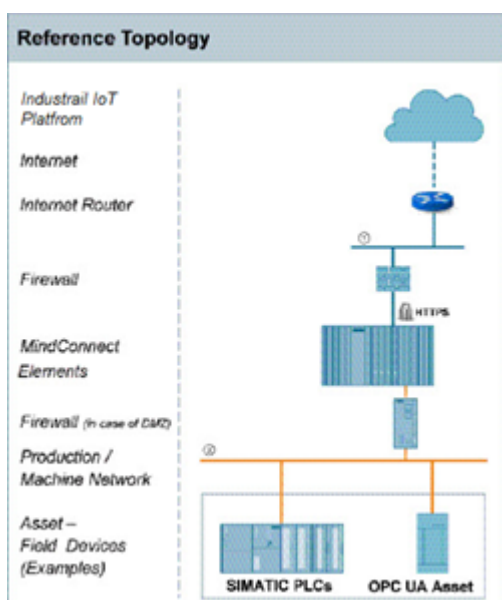
¹⁾ OPC UA @ PC always depends on the hardware.

- Read of a string data point you have to multiply the time with 2,5 per data point
- All devices are around 75% full

7.8 Firewall Settings

It is recommended to use a firewall between the Internet and the MindConnect Nano, this is also recommended for communication to the automation network.

In the direction to Automation Network a firewall supporting NAT (in case of DMZ, see section "List of abbreviations") or supporting the "Ghost-Mode" is required. Siemens offers many types of Firewalls for fulfilling these requirements.



① Corporate / Office Network with route to the internet or direct internet access, e.g. via a DSL modem

② Production / Machine Network

“Ghost-Mode”, also known as “Transparent Mode”, is used to protect individual, event alternating, devices by dynamically taking over the IP address.

Firewall/Proxy rules for MindConnect Nano

MindConnect Nano require open HTTPS and DNS ports for communication with Industrial IoT. You can open port 443 to enable this.

To communicate agent only, you will only need the `.eu1.mindsphere.io`.

7.9 Return of defective hardware

In case of a defect or corrupted hardware you can find contacts and information under the following links:

Spare part services

In case of a defect or corrupted hardware you can find detail information in the [SIOS Article - Spare Parts Services](#).

Delivery of spare parts

For ordering, provision and delivery of spare parts you can find more information in the [SIOS Article - Delivery of Spare Parts](#).

Siemens Contact Database

You can find your personal contact for spare parts in the [Siemens Contact Database](#).

7.10 List of abbreviations

Abbreviation	Description
DMZ	Demilitarized Zone (referred to a perimeter network)
GUI	Graphical User Interface
HTTPS	Hyper Text Transfer Protocol Secure
IoT	Internet of things
IoT2040	MindConnect IoT2040
IP	Internet Protocol
LAN	Local Area Network
MCN	MindConnect Nano
MSU	Industrial IoT Unit
NAPT	Network Address and Port Translation
OEM	Original Equipment Manufacturer
OPC UA	Open Platform Communications Unified Architecture
PC	Personal Computer
S7	STEP 7
UI	User Interface

7.11 MindConnect EthernetIP - CIP Communication path (route parameters) settings

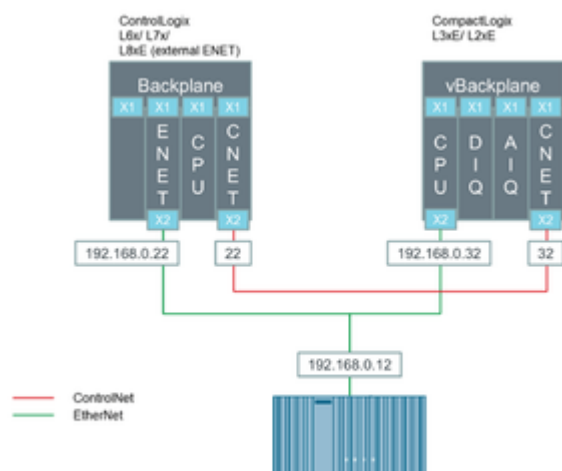
The CIP uses a method of encapsulating the messages adding communication path segments to the command segment. This leads to a longer and longer growing message as the number of

routing stations grows.

As an example, the following simple layout is used to explain that fact.

The MindConnect is acting as a CIP client.

The CIP server is provided by a ControlLogix or CompactLogix.



Each CIP message contains the routing information to the target. Every time a bridge accepts such a CIP message it strips out its own address information and forwards the modified CIP message to the next addressee along the way. When there is no further routing information available the destination target has been reached and the CIP message will be interpreted. Once interpreted the response will be send the same way back the request came. This means each bridge keeps a reference to the request package.

7.12 Case 1 – Accessing data inside an Allen- Bradley PLC

In this case, a ControlLogix uses the ENET bridge in slot 1 (0 based counting) for communicating with the CIP client. The processor module itself is in slot 2. An additional ControlNet Bridge is in slot 4 and is not used in this scenario.

The MindConnect sends a request to the ControlLogix on 192.168.0.22, slot 2.



This communication path (route parameters) is formed as follows:

It recognizes the CIP message is not for it and therefore, it removes its address part (1) from the CIP message. The remaining CIP message is being forwarded to the processor module as indicated by the remaining address information (2). The message contains a reduced communication path with the slot number. The processor removes that information and start interpreting the remaining message, as there is no further routing information in the packet. The following Wireshark packet analysis shows the additional segment.

7.13 Case 2 – Accessing the Allen- Bradley PLC via integrated Ethernet port

In this scenario, the CIP client sends some data into the CPU module using the integrated Ethernet port. In a CPU module with integrated Ethernet port, the Ethernet port is in the same slot as the CPU module.

The CIP client in the MindConnect sends a request package to the CPU module at 192.168.0.32 addressing the processor module in slot 0.



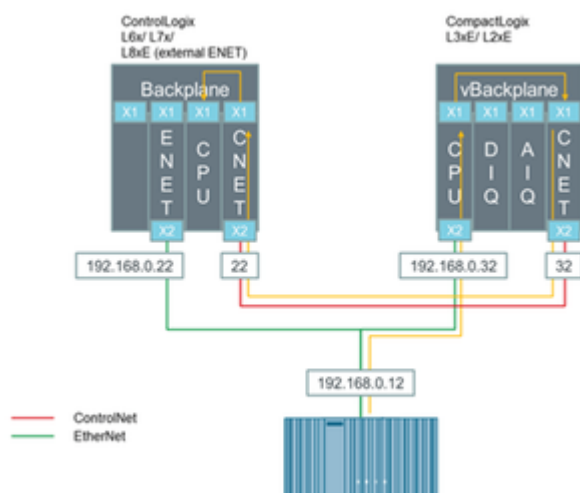
Following the previous schema, the above image represents the communication path (route parameters).

Following the previous made explanations, the CIP message travels first through the processor module to the backplane and it returns back to the processor module for interpretation. For such situations, it is possible to use an abbreviated communication path only containing the IPv4 address.

The MindConnect block uses the parameter "slot" to form the communication path (route parameters).

7.14 Case 3 – Accessing the Allen- Bradley PLC via the ControlNet interface

In this scenario, the CIP client reads data from the Allen- Bradley PLC. Now, the communication path will be through the CompactLogix acting as bridge between EtherNet/IP and ControlNet.



The yellow line in the above image represents the communication path (route parameters). In the communication path, the red marked sections always contain a network specific address. The blue marked part identifies the backplane or the interface to be used. It is not important that the backplane is physically existing or virtual. The green part identifies the slot number of the next bridge module (next hop) or the destination module.

7.15 MindConnect General Security Principles

The following are the security principles of MindConnect services:

- Communication between devices and Industrial IoT cloud services through HTTPS and signed URLs.
- Authentication and authorization using OAuth 2.0 mechanisms.
- Security tokens have a limited time span of validity and must be renewed.
- Before installation, firmware is signed and signature is validated by the device.

Onboarding agent

To onboard an agent, establish the trust between agent and agent management:

1. Create an agent in Asset Manager.
2. Define the network and security profile in Asset Manager.
3. Download the initial access token and network configuration to your local machine.
4. Deploy the initial access token to agent, for example via USB stick.

With the initial access token, the agent can prove its identity towards agent management by:

- The Initial Access Token (IAT) is physically entitled to onboard the agent.
- The initial access token is valid for 7 days for security measures to avoid the misuse.

Registering an agent

To register an agent at agent management, follow the below OAuth 2.0 protocol:
MindConnect Elements uses RSA3072 standard public/private keys in further communication.
For example, the agent registers at agent management with the IAT and additionally, a JSON Web-Key is set according to RFC7517 that contains a self generated public key.

Accepting the registration

With this response, the agent is onboarded and agent management accepts the registration.
There are two cases, the agent management responds with:

Case 1:

- The shared secret (valid for 7 days) – to be used to access token in the request.
- The re-registration access token (RAT) – to be used for the renewal of registration, once the registration is expired.

Case 2:

- The re-registration access token (RAT) – to be used for the renewal of registration, once the registration is expired.
- The agent management stores the public key received after registering an agent and uses it to verify the request access token from agent, which uses JWT signed with private key from now on. For more information, see [Onboarding an Agent](#).

Validity of the credentials

By default, each registered agent (either RSA 3072 or Shared Secret security profile) has its own credentials valid for 7 days and it must be renewed on a regular basis. After this period, Industrial IoT will not grant any access tokens to the agent and the agent credentials will be invalid. The agent needs to re-register by providing its Registration Access Token (RAT). Registration access tokens have no practical expiration times but each time a client refreshes its registration a new RAT is generated by Industrial IoT.

API communication agent

Except the one for onboarding and registration, all Industrial IoT APIs require an access token:
The Agent requests an access token from Agent IAM using a self signed JSON Web Token (JWT).
The JWT is signed (see RFC-7515) with either the shared secret (Case 1) or the private key of the

public / private key pair (Case 2) in Step 2. The JWT contains the tenantId

The granted access token is valid for 30 minutes. It is also a JWT and contains:

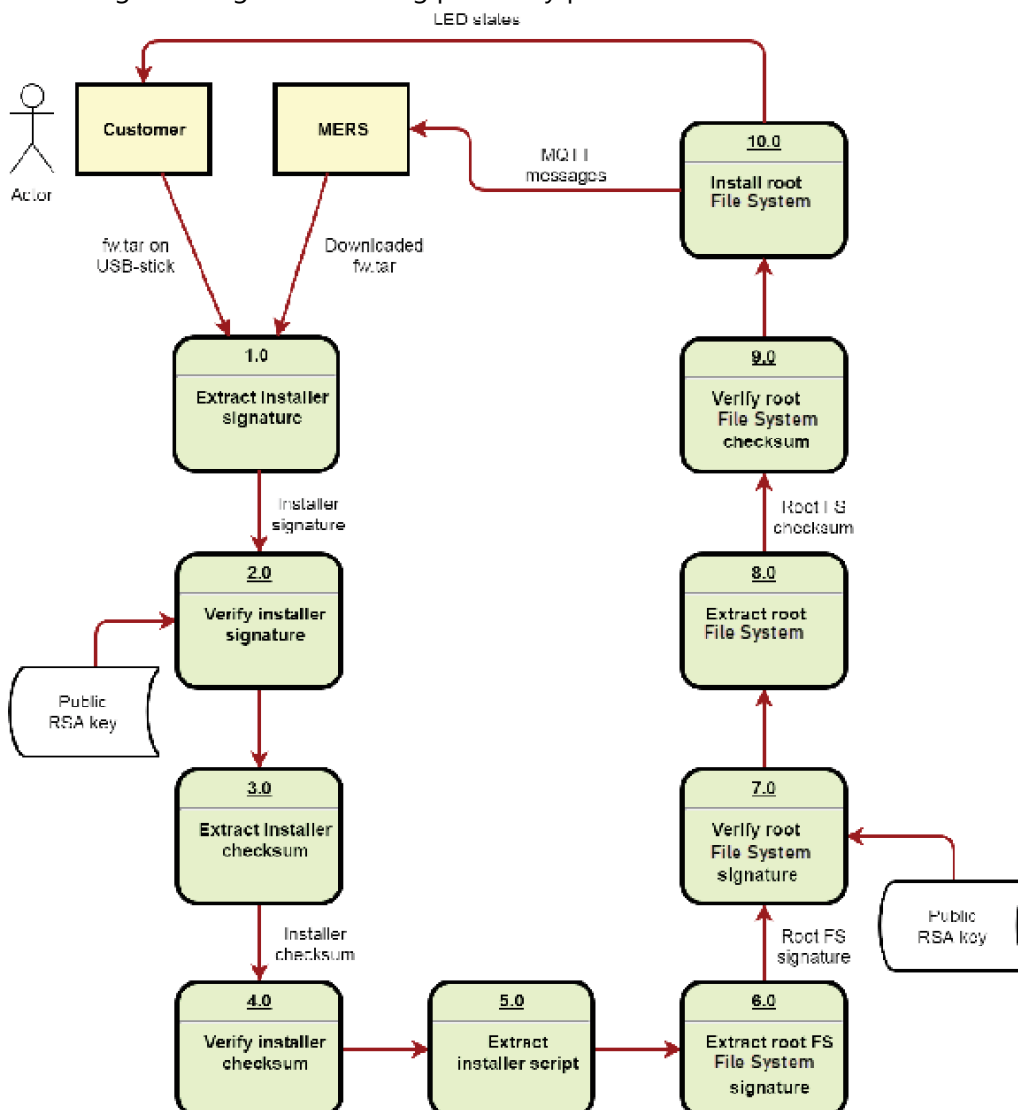
- tenantId
- scopes: It describes the type of services used in this token holder.

After expiry, a new token needs to be requested.

Firmware management

Firmware management is stored in secure cloud storage. Agent accesses secure storage through a signed URL issued by Industrial IoT and it is valid for 7 days.

Firmware is signed with private key from Industrial IoT. Firmware installer (MERS) on device, checks signature against matching public key provided at initial installation in the factory.



7.16 Default values for datasource Import and Export

The below mentioned table provides you the default values for the datasource import and export.

Protocol	Type	Column	Default Value
S7	Datasource	Connection Type Rack Number Slot Number	Automatic 0 2
OPCUA	Datasource	AuthenticationType Security Mode PLC Type	None None S7-1200
MODBUS-TCP	Datasource	Port Byte Timeout Response Suspension Time Request Delay	502 500 500 0 0
	Datapoint	Slave Number Start Address Quantity Variable Type Integer Byte Order Response Address Offset	1 0 1 Unassigned short Normal order 0
MODBUS-RTU	Datapoint	Slave Number Start Address Quantity Variable Type Response Address Offset Response Quantity	1 0 1 Unassigned short 0 1
Sinumerik	Datasource	Slot Number	2
Bacnet	Datasource	Address Binding	Dynamic
All protocols	Datapoints	On data change Hysteresis	False 0

8.1 Glossary

Aspect

Aspects are a data modeling mechanisms for assets. Aspects group the data points based on logical sense. For example: The pump skid has an aspect e.g. "Energy_consumption" that contains the data points: "power", "current", "voltage" etc.

Aspect is specified in Asset Manager and its name can be freely chosen, but should bring together a logical grouping of data points and a physical asset.

Asset

An asset is a digital representation of a machine or an automation system with one or multiple automation units (e.g. PLC) connected to Industrial IoT.

Data collection and data provisioning is based on so called (virtual) assets. This can be anything such as a pump, motor, PLC, an entire tool machine, a production line, a robot, a crane, a car, a wind turbine and so on. The data of an asset is collected and sent to Industrial IoT to make that data available for further processing and analytics.

Asset Identifier

Asset Identifier is an identifier assigned by the manufacturer of the device, to which MindConnect Nano will be connected. After the onboarding process, MindConnect Nano is connected to this device. The serial number identifies the asset to which the data belongs.

Asset Manager

Asset Manager is a Web Graphical User Interface for asset configuration. According to these configurations, the following functions are available:

Asset Configuration: assets can be created, onboarded, modified, moved, offboarded or deleted.

Asset type

Asset type is a sort or kind of a product line made by one manufacturer.

Components

Components represent built-in functionality. Components are:

- Asset Manager
- Insights Hub Monitor
- Settings
- UTC Reporting

For the complete list and description of Components, refer to the documentation.

Data point

Data points refer to elements (variables), which allow values to be obtained from data sources (OPC UA or S7 etc.). They are combined into a relevant aspect. For example, "temperature" and "torque" are data points of an aspect "Energy_consumption".

Data points are configured in "Asset Manager". In "Insights Hub Monitor", their values are visualized as time series.

Data source

A data source is a physical element of a device, which can be monitored by Industrial IoT.

For example: OPC UA Server, S7.

Event

In "Insights Hub Monitor", an event is a change of a datapoint state. Events are used for the requests. With a rule it is possible to define the request, which will be created in Industrial IoT when the event is triggered.

Besides the monitoring rule, a description of event (e.g. "The limit is exceeded, this may indicate damage in the pump"), the resulting action (e.g. "Please contact your hotline") and priority (urgent, important or info) can be lodged.

See also "request", "rule".

Last Connection

"Last Connection" status in "Asset Manager" gives the information, when a MindConnect Element was online for the last time.

MindConnect Element

MindConnect Elements are devices for transferring data and allow connectivity to Industrial IoT such as MindConnect Nano, MindConnect IoT2040 or MindConnect FB.

Industrial IoT

Industrial IoT is the Siemens Industrial IoT operation system comprising the core cloud services and applications, whereas the MindConnect Element provides secure and easy connectivity from the field or machine to Industrial IoT. In Industrial IoT, submitted data by a MindConnect Element is processed and stored for analysis and further management purposes. For more information, refer to the documentation.

Launchpad

Launchpad is the entry point for components as well as for available applications.

Online/Offline Mode in Asset Manager

Online indicates internet connection, when the MindConnect Nano/IoT2040 sends the data to the server. Offline mode means the absence of the internet connection and the MindConnect Nano/IoT2040 is no longer able to send data to the server.

Insights Hub Monitor

Insights Hub Monitor is an easy-to-use basic analytics application of time series data that enables the user to gain deeper insights into their production system and the machine performance. It uses the assets, aspects and variables as data model for its functions.

Organization

Organization is the part of hierarchical asset configuration in Asset Manager. Within organization, areas/assets are specified.

As the administrator (OEM), you manage organizations, their customers and assets. As the user of an organization, you can access the assets assigned only to this organization.

An organization can be anything and is defined by the administrator depending on their requirements: customer, cost center, branch etc.

Request

Requests are the messages which log asset incidents based on predefined rules.

You can create manual requests as well as requests for the regular acquisition of measured quantities with measuring series.

See also “event”, “rule”.

Rule

With rules you can configure an automatic monitoring of datapoints on events. For example, a request will be created automatically by the system, if a value exceeds or falls below a threshold (edge trigger).

See also “event”, “request”.

Settings

Settings is a Web Graphical User Interface for user, organization and role configuration.

Time series

Time series is a sequence of measurements which are produced by data sources over time. Analysis and visualization tools (e.g. Insights Hub Monitor) can retrieve collected time series and present it to the user after processing. In Asset Manager, the measurements, that have to be collected, can be specified.