vNode

User Guide

The Communication Platform for Industrial IoT
From the Edge to the Cloud

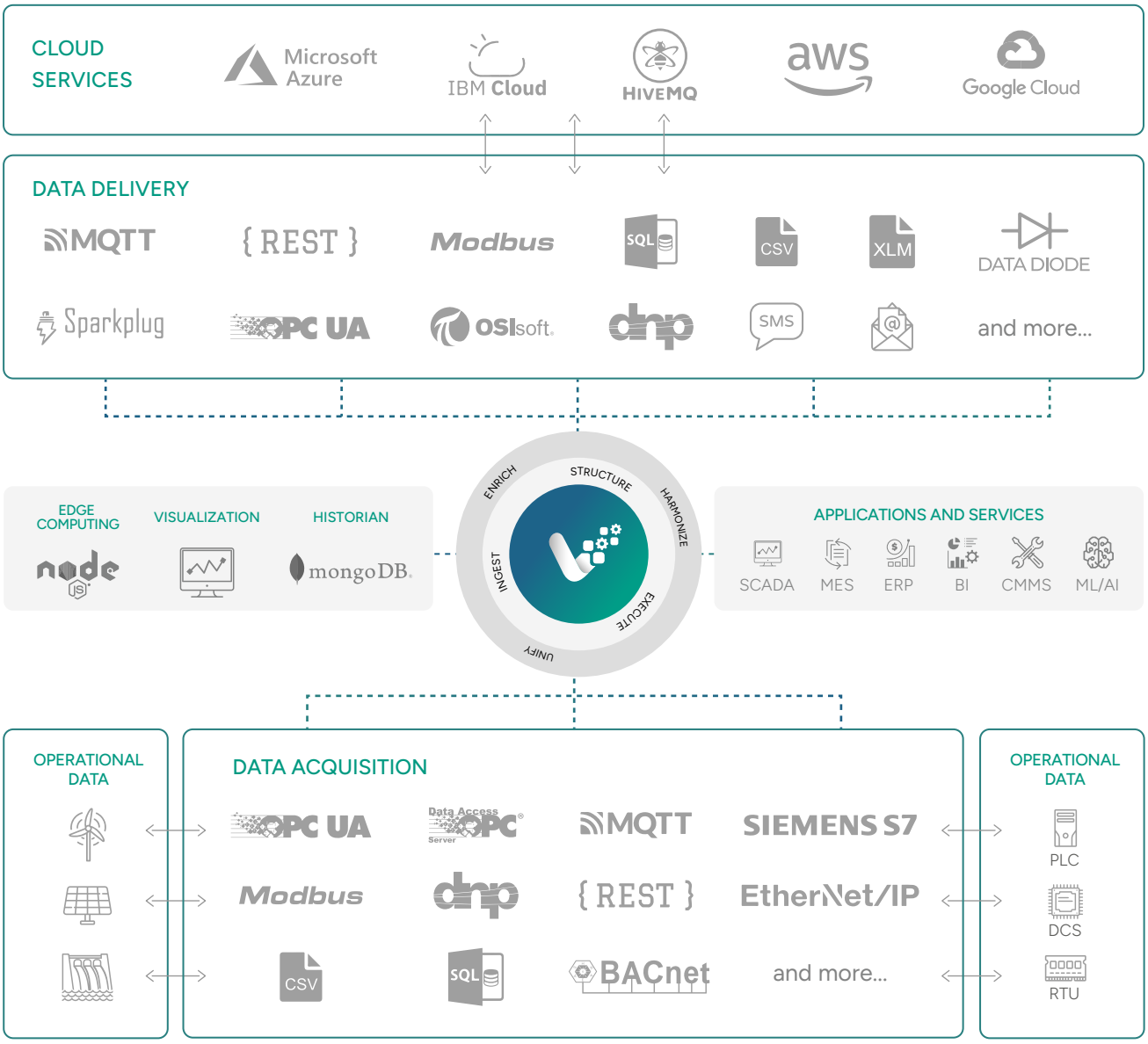
SECURE, RELIABLE AND SCALABLE.

vNode

Contents

Introduction

Welcome to **vNode**, the next generation of accessible, scalable, data-centric IIoT and DataOps software. vNode was designed from the ground up to be user-friendly and easy to work with, while providing the flexibility and scalability required for complex, large-scale projects. This guide introduces vNode's architecture to help you quickly get started.



vNode is a complete **Industrial Edge Platform for IIoT & DataOps** that enables seamless integration between the industrial plant floor and third-party applications, whether on-premise or in the cloud.

With vNode, you can easily create bidirectional data pipelines between OT and IT systems and decouple devices from applications by consolidating, modeling, and processing all your operational data in a single source of truth and ultimately making all this data available across the entire organization.

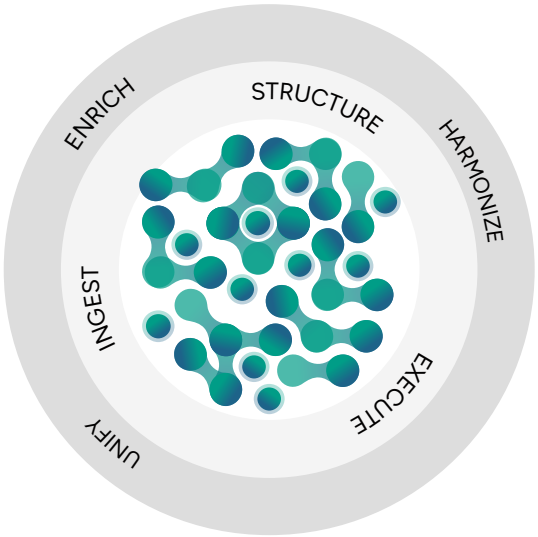
vNode Modules

vNode is a modular platform, meaning that its functionality can be adapted to the specific needs of each application using different modules. The following connectors allow data to be exchanged with other systems:

Data Acquisition
Collects data from a wide range of field devices and protocols, including PLCs, sensors and databases.

Data Management
Process and enrich your data with built-in features like alarms, calculations, historization and logic.

Data Delivery
Distribute processed data to cloud services, databases (SQL/MongoDB), SCADA systems and more.



Data Acquisition Modules (data collection and normalization)

Key modules include: Modbus Client, OPC UA Client, Siemens S7 Client, BACnet Client, Ethernet/IP Client, IEC 104/102 Client, DNP Client, MQTT Client, SNMP Client, SQL Client, REST API Client.

Data Delivery Modules (publishing and third-party integration)

Key modules include: OPC UA Server, REST API Server, Sparkplug Client, MQTT Client, Modbus Server, DNP3 Server, IEC 104 Server, Notifier, Data Exporter, Data Diode.

Edge & Visualization Modules (edge processing and visualization)

These modules process data locally at the edge, executes logic, stores historical data, and provide web-based HMIs for real-time operations. Key modules include Historian, Derived Tags, Scripting, WebVision, Redundancy Agent, Audit Trail, Linked Tags.

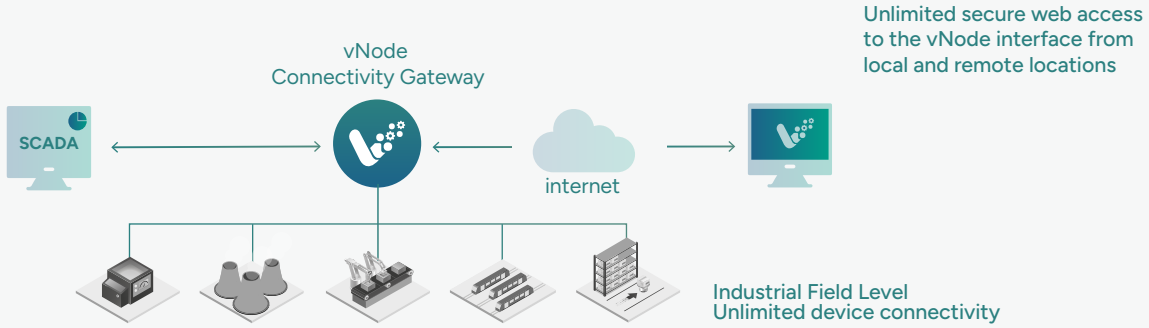
For the complete list of vNode modules and a detail description of each one, please refer to the vNode Module List or contact us at info@vnodeautomation.com

vNode Architectures

The following figures show different ways to use **vNode**

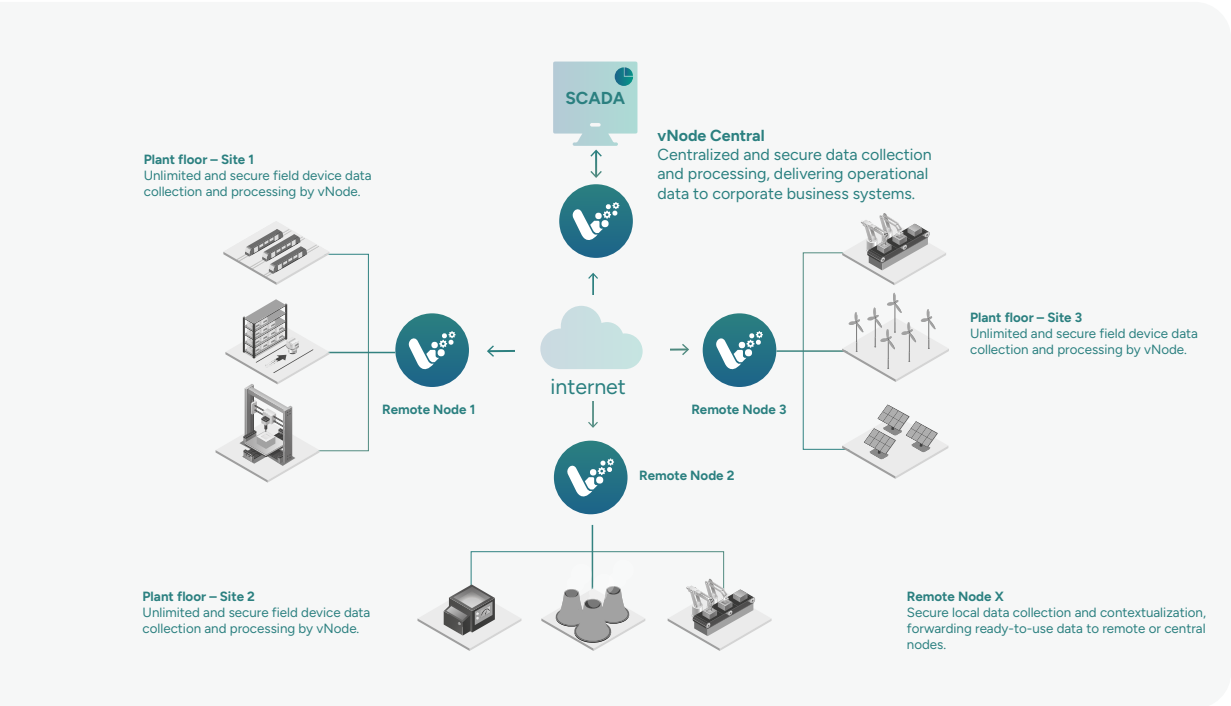
Secure Local Communications

Bridging Corporate and Control networks. This architecture provides a single-entry point to the plant floor in order to securely communicate with unlimited field devices from local SCADA systems and line-of-business applications.



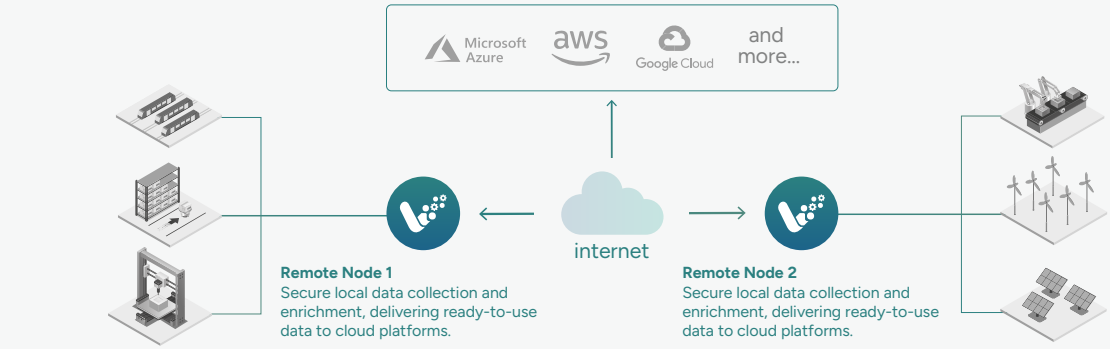
Reliable Wide-Area Communications

Built-in Store&Forward capability. Each **vNode** remote node acts as an Edge Gateway, collecting data locally and sending it to the central server using built-in Store&Forward mechanisms to ensure data integrity. All communications between **vNode** nodes incorporate built-in security mechanisms to ensure critical Industrial Cybersecurity.



Cloud Computing Platform

Plug-and-play connectivity to IoT Cloud Platforms. **vNode** instantly connects to any standard MQTT broker or to any main Cloud platforms such as Azure IoT Hub, AWS IoT and Google Cloud IoT. MQTT communications to IoT platforms include built-in unlimited Store&Forward capabilities and cutting-edge security mechanisms.



Remote Node 1
Secure local data collection and enrichment, delivering ready-to-use data to cloud platforms.

Remote Node 2
Secure local data collection and enrichment, delivering ready-to-use data to cloud platforms.

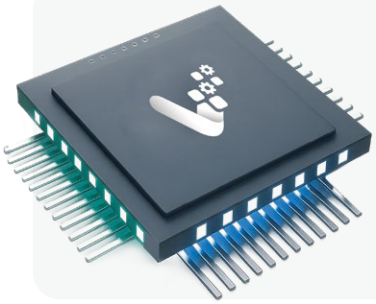
Plant floor – Site 1
Unlimited field device connectivity with secure data processing powered by vNode.

Plant floor – Site 2
Unlimited field device connectivity with secure data processing powered by vNode.

Installation and Upgrade

Getting vNode up and running is quick and easy. Installation takes less than a minute and the system will then be ready to immediately start collecting data. Simply download the installer from the vNode website, run the installer and the WebUI will automatically open as soon as the installer has finished.

Windows Setup



vNode 1.22 is compatible with the following Windows versions:

- Windows 8, 10 and 11
- Windows Server 2012, 2012 R2, 2016, 2019, 2022 and 2025
- Windows Server Core
- Windows IoT Enterprise 10 and 11

Minimum system requirements:

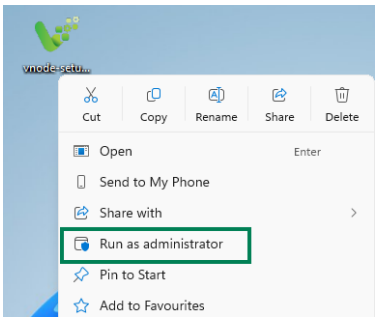
- Single-core 1GHz processor.
- 1GB RAM.
- 1GB free HD space (Requirements vary by usage).

Install vNode

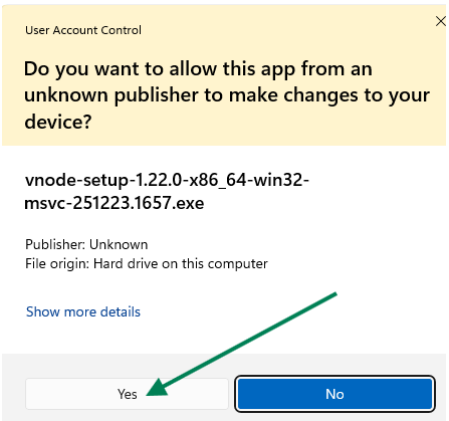
Manual install

Setup 1: Download the Windows x64 setup from our Downloads page.

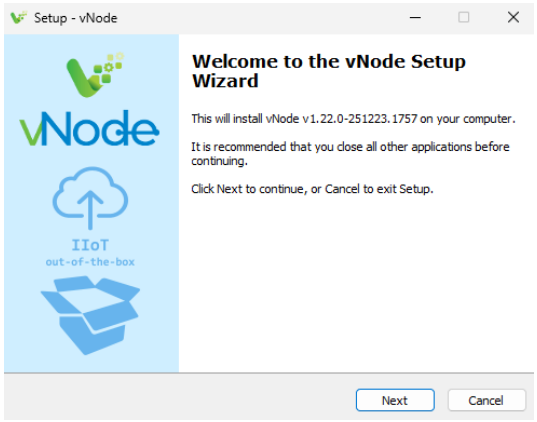
Setup 2: Right-click on the setup file and select “Run as administrator”.



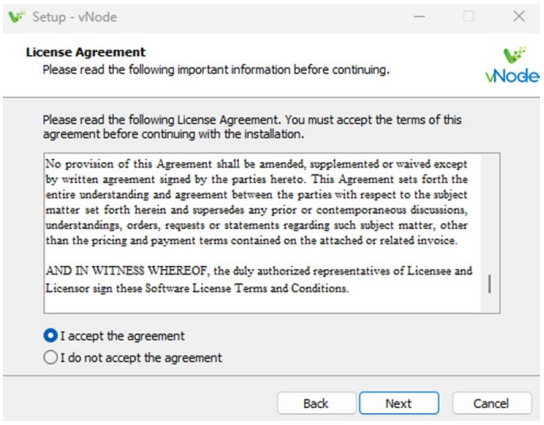
Step 3: Approve the User Account Control (UAC) prompt.



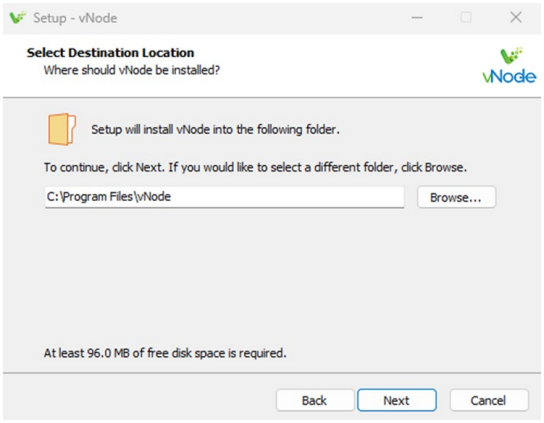
Click next:



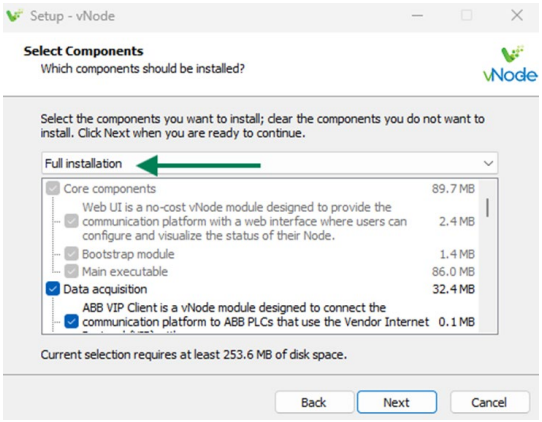
Step 4: Review and accept the vNode End User License Agreement.



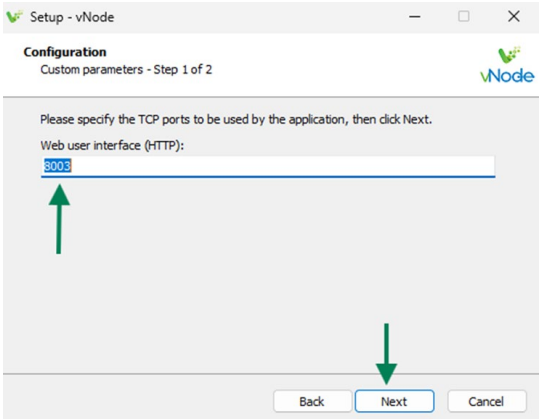
Step 5: Set the installation directory, which defaults to "C:\Program Files\vNode". This is the recommended location.



Step 6: Select the modules to be installed. We recommend choosing the *Full installation* option.



Step 7: Configure the default port to access the WebUI, ensuring this port is not in use by any other application.



TCP ports used by the vNode default installation:

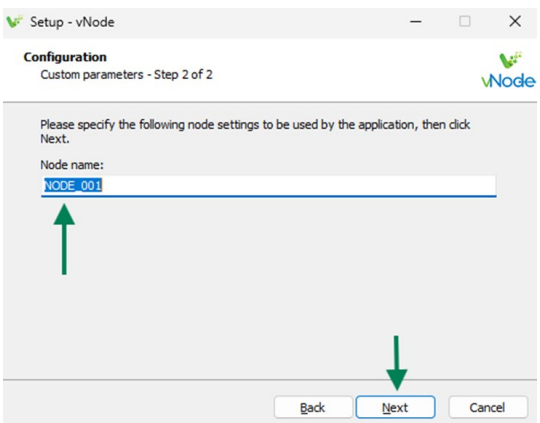
8003: Web interface (HTTP)

8443: Secure web interface (HTTPS)

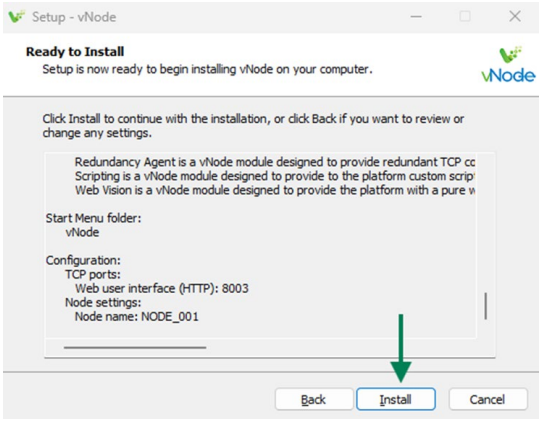
3002: Internal vNode communication

TCP ports assigned to vNode must not be in use by any other application.

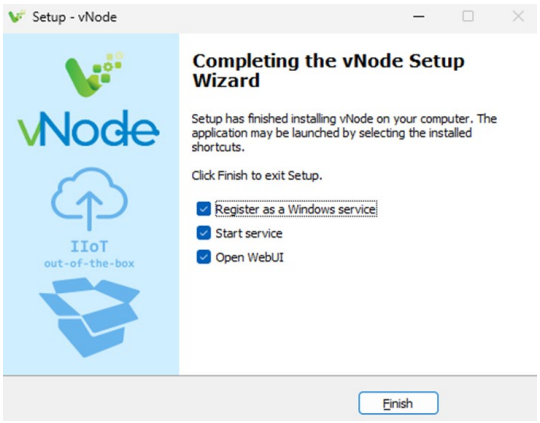
Step 8: Set a node name or leave the default, this name will be used as the node's unique identifier.



Step 9: Confirm the settings and proceed with the installation.

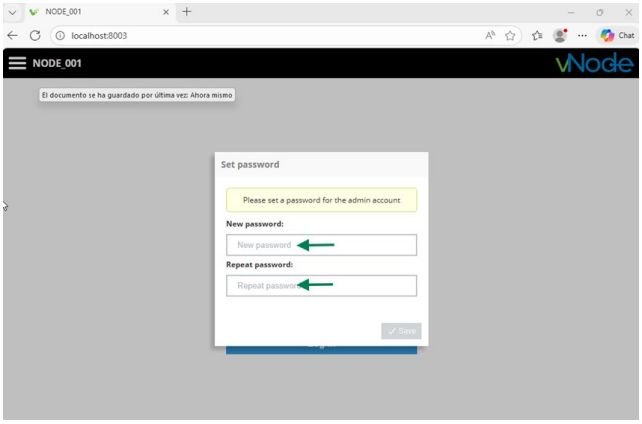


Step 10: Configure vNode to run as a Windows service allowing it to start automatically with the system and complete the installation process.



Step 11: The WebUI will automatically launch in your default web browser. You will be prompted to set the **admin** password during the initial login.

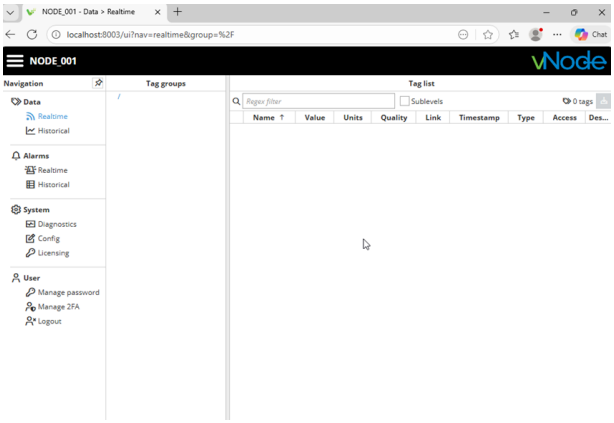
> To access vNode WebUI from a different machine, make sure that the Windows Firewall on the host machine is not blocking the port that was been assigned to vNode WebUI during setup.



User: admin	**Password:** set by the user

> In **demo** mode, vNode runs with full functionality for two hours. To restart the demo period, simply restart each module or service from the Web interface. The service may also be restarted from the Windows Start Menu or Windows Services ("vNode" service).

The vNode platform is now ready for configuration and operation.



vNode runs as a service and is automatically initiated when Windows is started. The service can also be stopped/started manually from the Start menu.

Upgrade vNode in Windows

> Before proceeding with an upgrade, ensure that the Support and Maintenance (S&M) of your license is active for the version you plan to install.

To update vNode on a machine running on Windows, follow these steps:

Step 1 (Recommended): Create a node backup to restore the **vNode** configuration if something goes wrong during the update. The steps to perform a node backup can be found in the **vNode Manual Backup and Restore** document.

Step 2 (Optional): Review the release notes for the target vNode version to identify any breaking changes that may affect your current configuration.

> It is strongly recommended to apply the update in a testing environment before implementing it in the production environment.

Step 3: Stop the **vNode** service by running the following command inside a **PowerShell (Administrator)** console or by using the Services management panel in Windows.

```
Stop-Service -Name "vNode"
```

Step 4: Proceed with the installation.

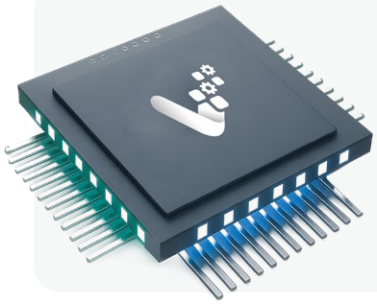
Starting with version **1.21**, vNode runs natively as a 64-bit application and will default to being installed in **C:\Program Files.** Be careful to modify this path in the installer when upgrading from previous versions to **C:\Program Files (x86).** Additionally, since version 1.21 includes substantial changes in managing Users and Roles, upgrading from a previous version **will automatically overwrite all previously configured users** for the WebUI and WebVision, requiring to re-create them.

Step 5: Log in to the WebUI to verify everything is running as expected.

Uninstall vNode on Windows

To uninstall vNode from a Windows device, click the 'Uninstall' option in the vNode Start menu.

Linux Setup



vNode 1.22 is compatible with the following Linux distributions:

- Debian 10 and newer; Debian-based like Ubuntu 20.04 and newer versions; Raspberry PI OS (64-bit).
- RHEL 7, 8 and 9; Red Hat-based like CentOS 7, 8 and 9; Oracle Linux 7, 8 and Amazon Linux 2, 2023.

Minimum system requirements:

- Single-core 1GHz processor.
- 1GB RAM.
- 1GB free HD space (Requirements vary by usage).

For other distros please contact your vNode distributor or email info@vnodeautomation.com

Install vNode on Linux

The following procedure will install vNode in the recommended /opt/vNode folder. It is also possible to install vNode in a different folder, depending on the end user's preferences.

Step 1: Download the vNode setup file for Linux from our Downloads page.

A Linux setup file for all the different distributions can be downloaded from www.vnodeautomation.com and uploaded or copied to the target machine.

Linux ARM64

cd / sudo wget https://vnodeautomation.com/downloads/latest/installer/linux-/vNode-setup-linux-ARM64.tar.gz

Linux x64

cd / sudo wget https://vnodeautomation.com/downloads/latest/installer/linux-/vNode-setup-linux-x64.tar.gz

Step 2: Decompress the setup file, where <arch> is the architecture of the target machine and <version> corresponds to the vNode version.

```
sudo tar -xvzf vnode setup-linux-<arch>-<version>.tar.gz -C /opt/
```

Step 3: Install vNode as a system service.

```
sudo /opt/vnode/bin/vnode install
```

In modern versions of 64-bit operating systems, external databases are not required to run Historian. A MongoDB instance will automatically be installed in the vNode folder to provide storage for historical data. The user can choose to use either the embedded MongoDB instance or an external one.

For older operating systems, such as Debian 9, RHEL 7 or Ubuntu 16.04, as well as ARM architectures, a MongoDB instance can be manually installed and used for Historian.

When installing MongoDB on an **ARM64** device, please note that starting from version **4.0**, MongoDB requires the **ARMv8.2-A** or later microarchitecture.

To use the embedded MongoDB database, it might be needed to install the **libcurl4** dependency using the following command:

```
sudo apt install libcurl4
```

Step 4: Start the vNode service.

```
sudo systemctl start vnode
```

Step 5: Log in to the WebUI at http://localhost:8003 (to access the WebUI from a different machine use the IP instead of localhost) using the default credentials shown in the table below. We recommend changing the credentials after the first login.

	Full access	Read-only access
user:	admin	user
password:	vnode	vnode

To access the vNode WebUI from a different machine, make sure that the vNode host machine is reachable and that there are no firewalls blocking the port assigned to vNode WebUI (8003 and/or 8443 by default).

In demo mode, each module runs with full functionality for two hours. To restart the demo mode, simply restart the module from the Web interface.

In demo mode, vNode operates with full functionality for a two-hour period.

To restart the demo period, perform one of the following actions:

- **Restart each module** directly from the WebUI.
- **Restart the service** through the Windows Services console (look for the "vnode" service).

The **vNode** service can be controlled from the console using the following commands:

```
# Start the service.
sudo systemctl start vnode

# Stop the service.
sudo systemctl stop vnode

# Restart the service.
sudo systemctl restart vnode

# Check the status of the service.
sudo systemctl status vnode

# View the logs of the service.
sudo journalctl -u vnode
```

Upgrade vNode in Linux

Before proceeding with an upgrade, ensure that the Support and Maintenance (S&M) of your license is active for the version you plan to install.

Step 1 (Optional): Create a node backup to restore the vNode configuration if something goes wrong duringthe update. You can find the steps to perform a node backup in the **vNode Manual Backup and Restore** document.

Step 2 (Recommended): Review the release notes for the target vNode version to identify any breaking changes that may affect your current configuration.

It is strongly recommended to apply the update in a testing environment before implementing it in the production environment.

Manual upgrade
To update vNode on a machine running Linux, follow these steps:

Step 1: Stop the vNode service by running the following command:

```
sudo systemctl stop vnode
```

Step 2: Extract the new version of the binary files from the setup, where <arch> is the architecture of the target machine and <version> corresponds to the downloaded vNode version. The following command assumes vNode is installed in /opt/vnode:

```
sudo tar -xvzf vnode-setup-linux-<distro>-<version>.tar.gz -C /opt/ ./vnode/bin
```

Given that version **1.21** includes significant modifications related to managing Users and Roles, upgrading from a previous version w**ill automatically overwrite all the users previously configured** for the WebUI and WebVision. Therfore, it will be required to recreate them. To ensure compatibility with the new version, the following command needs to be executed:

```
sudo tar -xvzf vnode-setup-linux-<arch>-<version>.tar.gz --skip-old-files
-C /opt/ ./vnode/config
```

Step 3: Start the service and log into the WebUI to check everything is running correctly:

```
sudo systemctl stop vnode
```

Uninstall vNode on Linux

Step 1: Uninstall the vNode service with the following command.

```
sudo /opt/vnode/bin/vnode uninstall
```

Step 2 (Optional): Delete the vNode installation directory

```
sudo rm -r /opt/vnode
```

This command will remove all the existing data and configuration of vNode.

Docker Setup

Docker is a platform that allows users to install and run **vNode** within a container. A container is a lightweight, self-sufficient environment that includes all the components and modules of **vNode.**

For a complete guide on how to install vNode on a docker please refer to the document vNode Docker Setup

Getting Started

The following steps allow users to configure a **vNode** node in order to collect data from field devices and share this data with other systems.

Step 1: Setup Modules to Activate Features

vNode functionalities are enabled using **modules.** In order to use a specific feature, the corresponding module must be installed during the setup process and activated in the configuration settings. For example, the WebUI is a module that is automatically activated, making it instantly accessible as soon as the installation is complete. Since vNode is a microservice-oriented architecture, each module runs as an independent process.

Bootstrap is the core service that manages the rest of the processes.

Active functionalities or modules also require licensing. Each module requires a valid license to run in production mode. If a module doesn't find a valid license, it will run in demo mode for two hours. In order to restart the demo period, the module must be restarted.

For more information about Modules refer to the document: vNode Module Instances

Step 2: Configuration of Data Source Modules (field connections)

This step involves configuring all connections with field devices and is only necessary for source modules that require field connections such as OPC UA client, OPC DA client, Modbus client, Siemens client, etc.

Step 3: Setup of Data Tags

In order to create tags, all main properties must be provided:

- **Data type/format**
- **Scaling and engineering units**
- **Data Source** (pointing to a connection configured in the previous step)
- **Alarms** (thresholds, deadbands, severity, delays, etc.)
- **Historization** (sampling, aggregation, storage policy)

Once the tag has been created and the configuration has been saved, the real-time value of the tag will be available from the **Real-Time** menu. If historization is enabled, you can also view historical data in the **Historical** view.

For more information about Tags refer to the document: vNode Manual Tags

Step 4: Edge Computing

vNode is built to process and contextualize data at the edge before exchanging it with other systems by applying logic close to the source.

- **Derived Tags:** Create computed tags from existing signals using low-code / no-code configuration (e.g., calculations, aggregation, statistical metrics/KPIs such as min/max/avg, normalization, redundancy handling, diagnostics).
- **Scripting:** Run event-driven automation and workflows in the node (e.g., timers, tag changes/conditions, system events) using JavaScript.

Step 5: Configuration of Data Destination Modules

vNode can output the collected data in several different ways:

- **Industrial servers:** OPC UA server, Modbus server, DNP3 server, and IEC 104 server.
- **MQTT:** Publish data to Azure, Amazon Web Services or standard MQTT brokers.
- **Sparkplug (Sparkplug B over MQTT):** Publish Sparkplug payloads so Sparkplug consumers—e.g., Ignition (via Cirrus Link)—can auto-discover and consume vNode data.• Historian and databases: Send data to Historian (MongoDB or TimescaleDB) or to a SQL database.
- **REST API server:** Provide real-time data, node status, and historical data.
- **MCP Server:** Standard Model Context Protocol (MCP) interface for AI agents to integrate with vNode programmatically.
- **OSISoft PI:** UFL connector to OSISoft PI (CSV files containing events).
- **File Output:** XML and CSV files (events and aggregated data), delivered to destinations such as an SFTP server or an AWS S3 bucket.
- **Critical Infrastructures:** Send data securely through a data diode in critical installations.

Step 6: Distributed Architecture (vNode Links)

vNode Links enable secure, bidirectional communication between vNode nodes, making it easy to build distributed architectures (multiple sites, edge-to-central, hub-and-spoke).

Key benefits include:

- **Store & Forward** for reliable delivery.
- **Encrypted communications** between nodes to protect data in transit.
- **Firewall-friendly connectivity** (typically outbound-only from remote sites).
- **Bandwidth-efficient** operation for constrained networks.
- **Time-consistent** data exchange (origin timestamps preserved).

See the document vNode Manual Links for more information about vNode Links.

Step 7: Web SCADA/HMI Visualization and Control (Web Vision)

Web Vision allows you to build web-based HMI/SCADA applications for **monitoring and control** on top of your vNode data model—from **plant-level SCADA** to **Layer 2** operational SCADA and centralized **control center** views.

Typical use cases include:

- Plant HMI/SCADA screens for operators (real-time monitoring, alarms, and control).
- Layer 2 SCADA for multi-area supervision and coordination.
- Control center dashboards aggregating data from multiple sites/nodes.

As a next step we recommend you to read about the WebUser Interface, please refer to the document

vNode Manual Web User Interface

vNode

info@vnodeautomation.com
www.vnodeautomation.com

USA: (+1) 754 755 0009 **UK:** (+44) 161 660 32416 **SPAIN:** (+34) 935 721 007 **COSTA RICA:** (+506) 222 523 44
FRANCE: (+33) 041 368 0106 **MEXICO:** (+52) 554 628 2593

4855 W Hillsboro Blvd STE B3, Coconut Creek FL 33073, USA