MGate MB3000-G2 Series
Modbus Gateway User Manual

Version 1.0, September 2025

www.moxa.com/products

MGate MB3000-G2 Series Modbus Gateway User Manual

Copyright Notice

Trademarks

Disclaimer

Technical Support Contact Information

www.moxa.com/support

Table of Contents

1. Introduction

Welcome to the new-generation MGate MB3000-G2 series of Modbus gateways. All models offer seamless Modbus TCP to RTU/ASCII integration and provide RS-232/422/485 ports for serial communication. One-, two-, and four-port models are available.

This user's manual applies to the following series:

- MGate MB3170-G2
- MGate MB3270-G2
- MGate MB3470-G2

Overview

The MGate MB3170-G2, MB3270-G2, and MB3470-G2 are the new generation of Modbus gateways that convert between Modbus TCP, RTU, ASCII, and even proprietary or extended Modbus serial protocols. These MGate gateways can convert protocols transparently or actively poll connected devices and cache the data in their memory. This allows the SCADA system to retrieve Modbus data directly from the gateway, instead of waiting for all Modbus devices to respond, greatly increasing communication performance.

Moxa developed the MGate Series using the IEC 62443-4-1 secure development life-cycle process. Adherence to the IEC 62443-4-2 design and guidelines ensures the secure connection and management of these gateways within industrial networks.

In addition, the user experience is enhanced by versatile installation options that accommodate diverse mounting and wiring requirements, a rugged design suitable for harsh environments, and easy-to-use configuration and troubleshooting tools for quick maintenance.

High-performance Modbus Conversion

MGate MB3170-G2/MB3270-G2/MB3470-G2 gateways operate in either transparent (direct protocol conversion) or agent (polling and catching) Mode. Transparent mode converts protocols on a per-request basis. However, if the Modbus system has a larger scale or there is a need for lower latency, agent mode significantly improves SCADA communication performance by allowing direct data retrieval from the gateway's cache.

Flexible Modbus Conversion for Various Systems

MGate seamlessly integrates Modbus TCP with Modbus RTU/ASCII in any combination, requiring no changes to existing systems. With multiple serial port models, the MGate simultaneously converts between Modbus serial and Modbus TCP, as well as serial-to-serial and TCP-to-TCP. Even with legacy serial, or COM port-based software, MGate's Real COM mode enables communication through Ethernet TCP with no software adjustments. Beyond standard Modbus RTU and ASCII, the MGate also converts proprietary and extended Modbus serial protocols, eliminating the need for custom development or additional equipment and ensuring near-universal Modbus system compatibility.

Intuitive Configuration and Minimal Maintenance

The MGate simplifies Modbus routing, allowing you to easily forward Modbus IDs to specific serial ports, IP addresses, or TCP ports. This simplified routing is especially beneficial in complex systems with unordered Modbus IDs and even eliminates the need to know the IDs being routed.

Downtime during maintenance can be costly. System warnings often require contacting vendors, resulting in lengthy troubleshooting as they gather information and consult with other parties. MGate MB3000-G2 gateways streamline this process by providing comprehensive troubleshooting information for rapid issue identification. One-select diagnostics gather all necessary data for technical support, empowering users to efficiently resolve issues and drastically reduce downtime.

Package Checklist

Each MGate MB3000-G2 Modbus gateway is shipped with the following items. When you receive your shipment, check the contents of the box carefully and notify your Moxa sales representative if any of the items are missing or appear to be damaged.

Standard Accessories

- 1 x MGate MB3000-G2 Modbus gateway with DIN-rail kit pre-installed
- Quick installation guide (printed)
- Warranty card

2. Getting Started

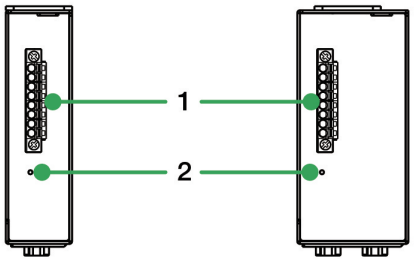
This chapter covers the hardware dimensions and installation of the MGate MB000-G2 Series. Software installation is covered in subsequent chapters, beginning with the "First-time setup" section.

Panel Layouts

Top View

MB3170-G2 Series
MB3270-G2 Series

MB3470-G2 Series



1. Chassis GND, dual power input, and relay output
2. Reset

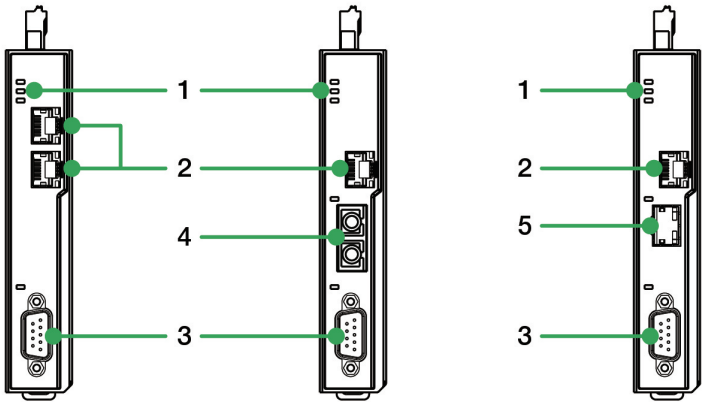
Front View

MGate MB3170-G2 Series

MB3170-G2 Series

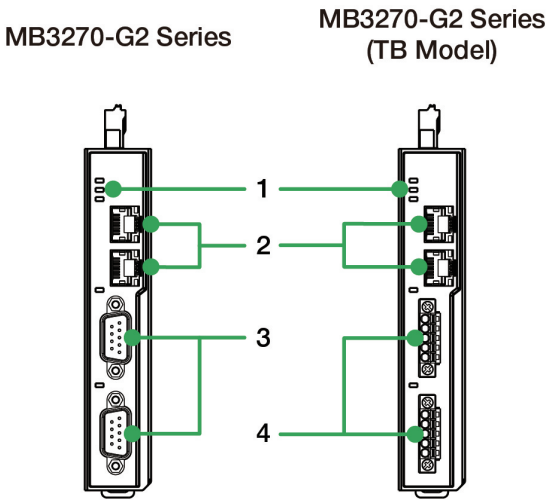
MB3170-G2 Series
(ST/SC Model)

MB3170-G2 Series
(SFP Model)



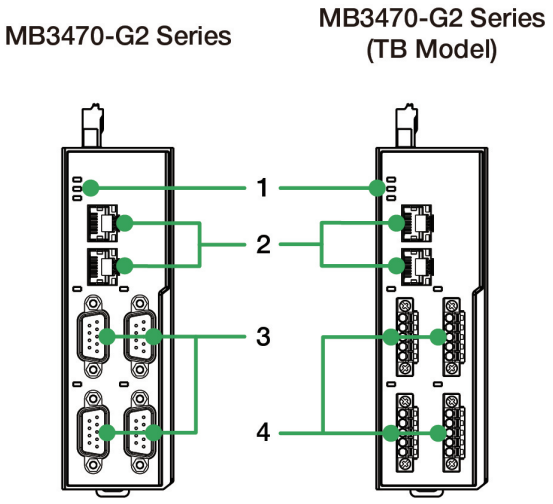
1. LED indicators
2. Ethernet port (RJ45)
3. RS-232/422/485
4. Ethernet fiber (ST/SC)
5. Ethernet fiber (SFP)

MGate MB3270-G2 Series



1. LED indicators
2. Ethernet port (RJ45)
3. RS-232/422/485 (DB9 male)
4. RS-232/422/485 (terminal block)

MGate MB3470-G2 Series

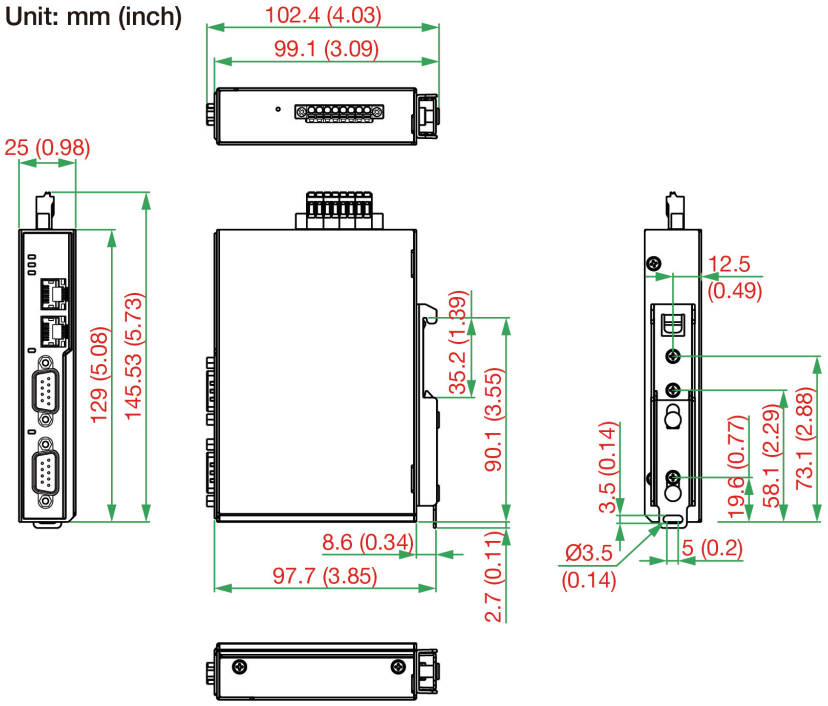


1. LED indicators
2. Ethernet port (RJ45)
3. RS-232/422/485 (DB9 male)
4. RS-232/422/485 (terminal block)

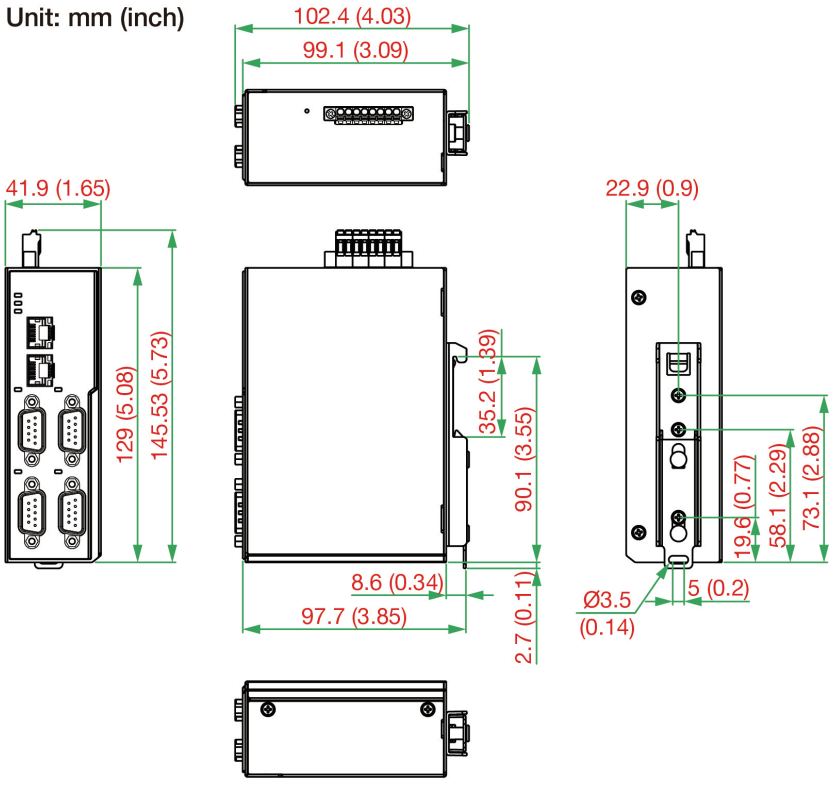
Dimensions

DIN Rail

MGate MB3170-G2/MGate MB3270-G2 Series



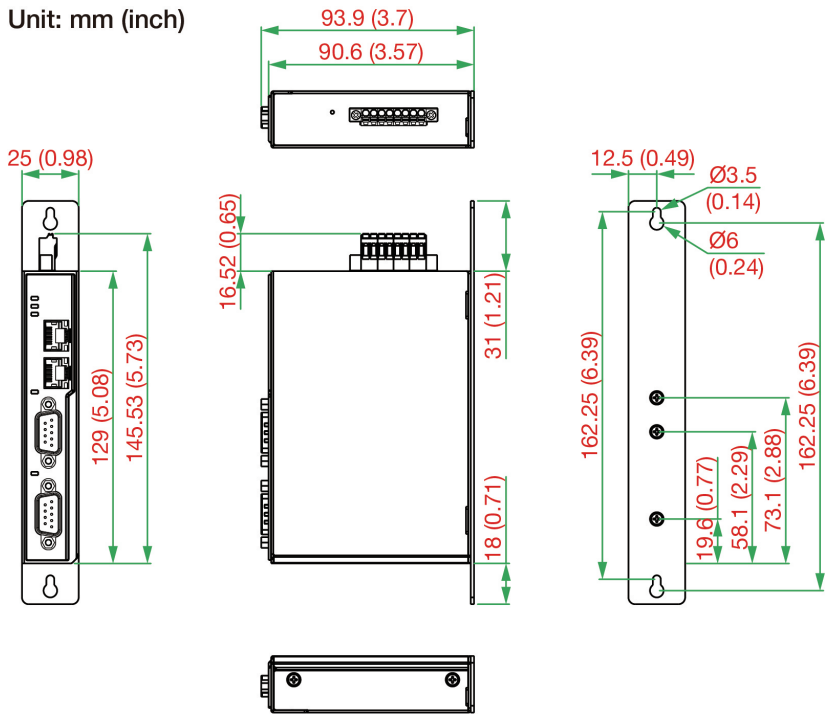
MGate MB3470-G2 Series



Wall Mount

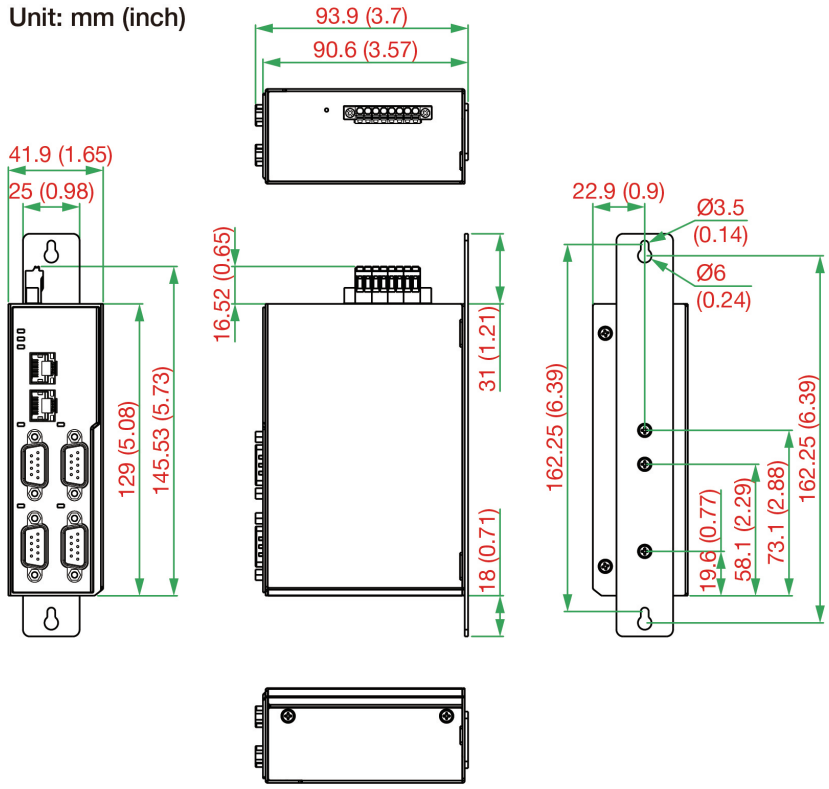
MGate MB3170-G2/MGate MB3270-G2 Series

Unit: mm (inch)



MGate MB3470-G2 Series

Unit: mm (inch)



Connecting Power

This section describes how to connect the power supply to the MGate MB3000-G2. Follow the instructions to connect a power source to the MGate's power terminal block:

1. Connect the 12 to 48 VDC power source or DIN-rail power supply to the MGate’s power terminal block.
2. Tighten the screws on both sides of the terminal block.
3. Turn on the power source.

NOTE

The unit does not have an on/off switch. It automatically turns on when it receives power. The PWR LED on the top panel will glow to show that the unit is receiving power.

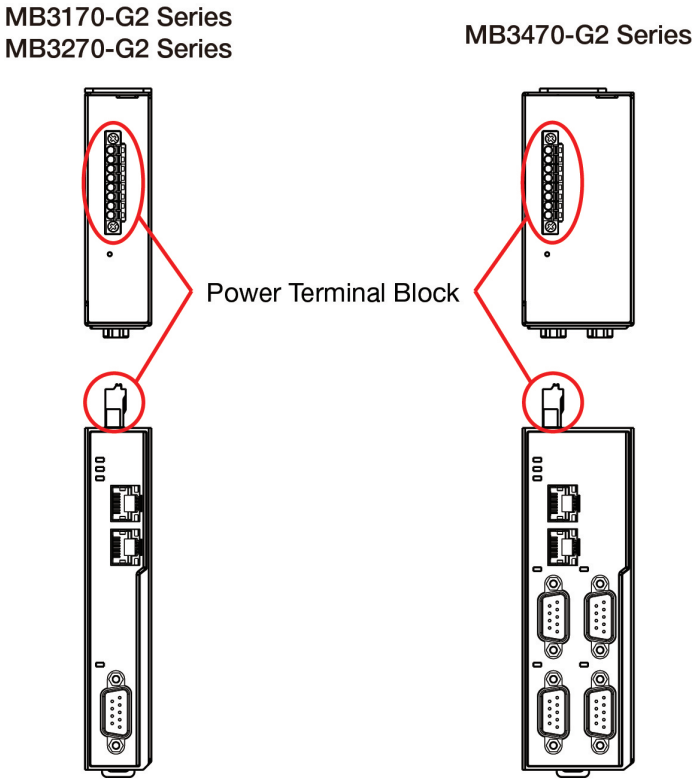
For power terminal block pin assignments, refer to the Quick Installation Guide, ***Power Input and Relay Output Pinouts*** section.

NOTE

Do not run communication wiring and power wiring in the same wire conduit. To avoid interference, wires with different signal characteristics should be routed separately.

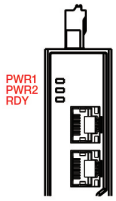
Powering Up the MGate MB3000-G2

Unbox the MGate and power it up. The following figures show the position of the power terminal block.

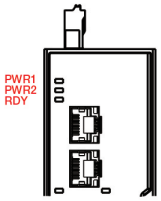


If you use a DIN-rail power supply or another vendor's power adapter, make sure the ground pin is properly connected. The ground pin must be connected with the chassis ground of the rack or the system. The Power (PWR) LED will change to green once the device is powered up. In a matter of seconds, the Ready LED will switch to green, and a beep will sound, signifying that the device is ready. The behavior of the LED indicators is outlined below.

MB3170-G2 Series
MB3270-G2 Series



MB3470-G2 Series



LED Indicators

The LED indicators on the front panel of the MGate MB3000-G2 are described in the following table.

Name	Color	Function
PWR1	Green	Power is on.
	Off	Power is off.
PWR2	Green	Power is on.
	Off	Power is off.
READY	Red	Steady: Power is on, and the unit is booting up.
		Blinking: IP conflict, DHCP server did not respond properly.
	Green	Steady: Power is on, and the unit is functioning normally.
		Blinking: The device server has been located by the "Locate" function from Moxa software utilities.
	Off	Power is off or power error condition exists
LAN1, LAN2 (LAN1 could be a fiber port, depending on the model)	Yellow	Steady on: Ethernet cable is connected, and the MGate detects it's linked up with 10 Mbps.
		Blinking: Indicates there is traffic on the Ethernet port.
	Green	Steady on: Ethernet cable is connected, and the MGate detects it's linked up with 100 Mbps.
		Blinking: Indicates there is traffic on the Ethernet port.
	Off	The Ethernet cable is not connected.
P1, P2, P3, P4	Yellow	Blinking: Serial port is receiving data.
	Green	Blinking: Serial port is transmitting data.
	Off	Serial port is not transmitting or receiving data

When the device is ready, connect an Ethernet cable to the MGate MB3000-G2 directly with the computer's Ethernet port or an Ethernet port of a switch.

To connect the serial device to the serial port of the MGate MB3000-G2, follow the pin assignment below.

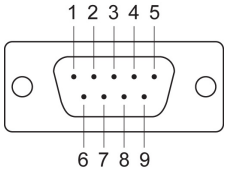
Connecting Serial Devices

The unit's serial port(s) are on the front panel.

> **NOTE**
>
> All device(s) that are connected to a single serial port on the MGate must use the same protocol (i.e., Modbus RTU, Modbus ASCII, or Modbus extension/proprietary protocols).

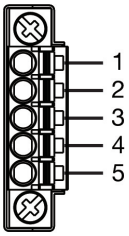
For serial port pin assignments, refer to the following tables.

RS-232/422/485 pin assignment (male DB9):



Pin	RS-232	RS-422/RS-485 (4W)	RS-485 (2W)
1	DCD	TxD-	–
2	RxD	TxD+	–
3	TxD	RxD+	Data+
4	DTR	RxD-	Data-
5	GND	GND	GND
6	DSR	–	–
7	RTS	–	–
8	CTS	–	–
9	–	–	–

RS-232/422/485 pin assignment (5-pin terminal block, for MGate -TB models only):

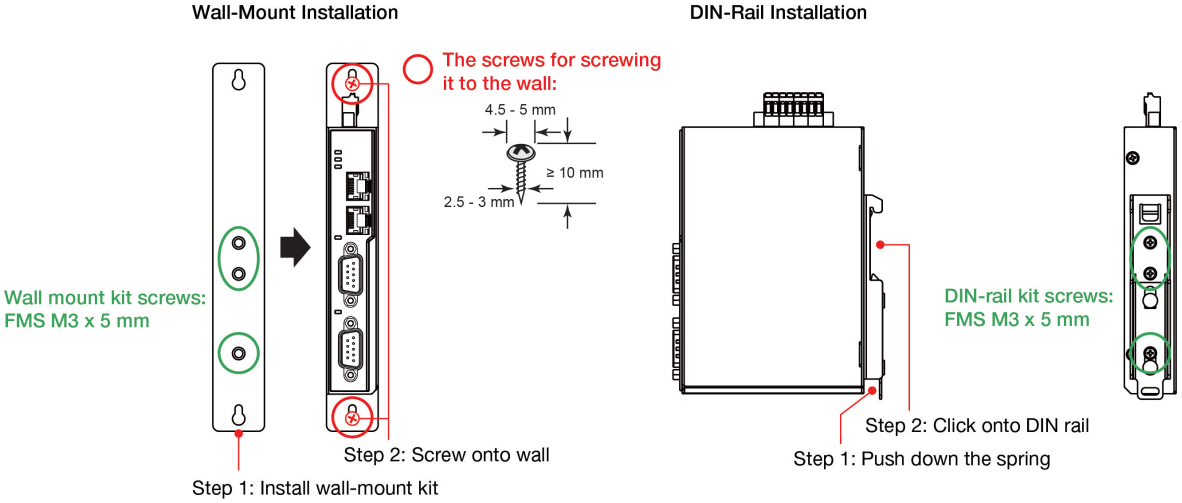


Pin	RS-232	RS-422/RS-485 (4W)	RS-485 (2W)
1	Rx	T+	–
2	–	T-	–
3	Tx	R+	Data+
4	–	R-	Data-
5	GND	GND	GND

For RS-485 terminators or pull high/low resistors, it is enabled by software. Refer to the ***Serial Parameters*** section in the following chapter.

Mounting the Unit

The MGate MB3000-G2 comes with a DIN-rail mounting kit. There are options to order a wall-mount kit for different placements. See the following diagrams for mounting instructions.



For the detailed screw length, refer to the Quick Installation Guide, ***Hardware Installation Procedure*** section for the complete information.

Connecting to a Host or the Network

Connect one end of the Ethernet cable to the MGate MB3000-G2's 10/100M Ethernet port and the other end of the cable to a host or the Ethernet network.

If the cable is properly connected, the MGate MB3000-G2 will show a valid connection to the Ethernet:

- The Ethernet LED glows solid green when connected to a 100 Mbps Ethernet network.
- The Ethernet LED glows solid orange when connected to a 10 Mbps Ethernet network.
- The Ethernet LED flashes when Ethernet packets are being transmitted or received.

Also refer to the LED indicators table for details.

3. First-time Setup

The MGate MB3170-G2, MB3270-G2, and MB3470-G2 are the new generation of Modbus gateways that convert between Modbus TCP, RTU, ASCII, and even proprietary or extended Modbus serial protocols.

These series are developed using the IEC 62443-4-1 secure development lifecycle process. Adherence to the IEC 62443-4-2 design and guidelines ensures the secure connection and management of these gateways within industrial networks.

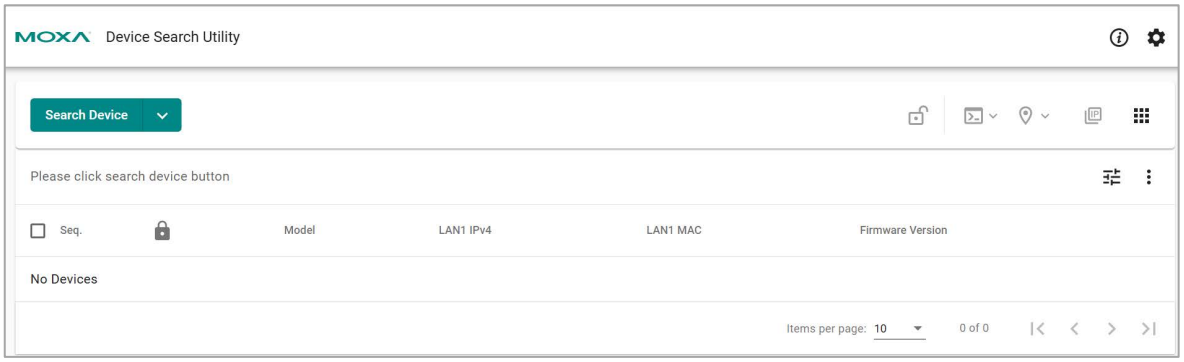
To accomplish this security level, the unsecured services will be disabled until you set up the first username and password for the unit. After logging in, services can be enabled as needed.

NOTE

For the MGate's configuration, the unit can only be configured and made functional using a web console (HTTPS) or Moxa service (through Moxa utilities).

Finding the Device

The default IP address of each MGate MB3000-G2 Series is https://192.168.127.254. Directly input the IP address at the address bar of a browser to open the web console to set up the first username and password. Or download the **Device Search Utility (DSU) v3.0** and search for the device to access its web console.



DSU is a handy tool for easily finding MGates and deploying single or multiple devices. DSU v3.0 functions as a web-based application that works on Chrome, Firefox and (Microsoft) Edge. To use the web-based application DSU v3.0, your browser version and operating system must meet certain minimum requirements:

- Chrome:
  - For Windows 7, 8/8.1, Server 2012 and Server 2012 R2: Chrome 109 and newer
  - For Windows 10 and newer, Server 2016 and newer: All Chrome versions
- Firefox:
  - For Windows 7 and newer versions, Server 2012 and newer versions: All Firefox ESR versions
- Edge:
  - For Windows 7 and newer versions, Server 2012 and newer versions: All Firefox ESR versions

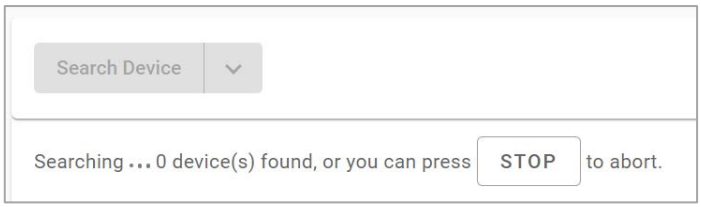
NOTE

For detailed instructions on DSU, download the user manual from moxa.com.

Search Device

When connecting the MGate to the network, the DSU's Search Device function can be used to find the target MGate. Searching can be done in three different ways. To see the options, select on the pull-down menu:

Searching Method	Description
Search	Default button action. It will search the devices by multicasting.
Search by IP	Search the device by a specific IP
Search by IP range	Search the device in a certain IP range; the search results will only display the corresponding IP type. For example, if you search by IPv4, only IPv4 values will be displayed.



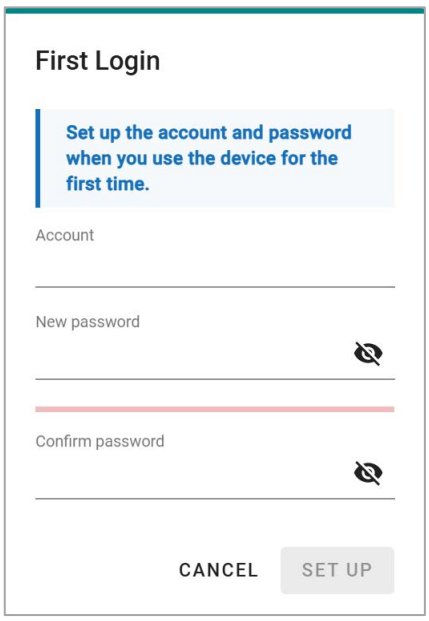
You can stop the search at any stage in the process. Select the **STOP** button on top of the table to halt the search and keep the already searched devices on the list.

The default search time is 10 seconds. DSU will continue searching until time runs out. If your device(s) does not appear, you may change the search timeout limit in **Preferences > Device Search > Timeout limit for device searching**, to give the network a bit more time to respond.

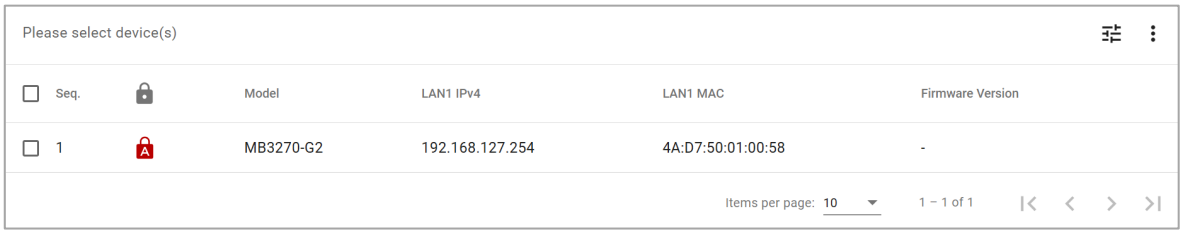
First-time Login With Device Search Utility (DSU)

To address cybersecurity concerns, the MGate found through DSU will prompt for an account name and password during the first login. Select the target device, meaning the factory default device, and select the unlock button .

The login window will remind you to set up the account name and password, and it will show the password minimum requirements as tips below the password field.

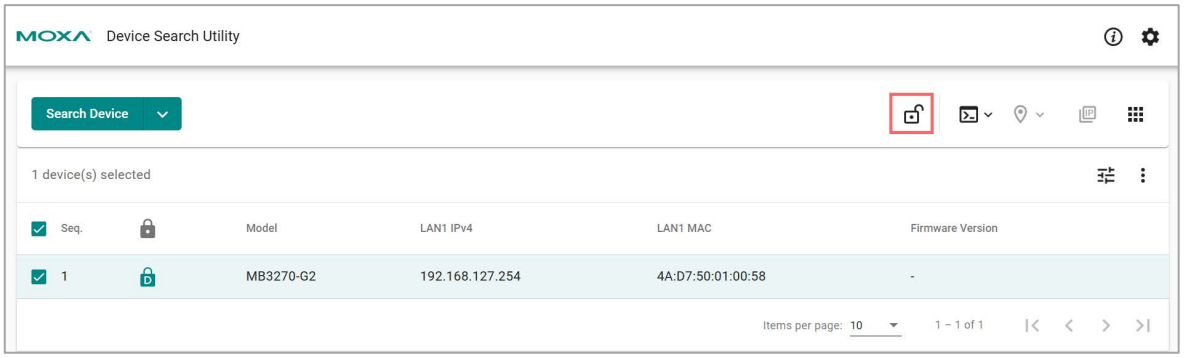


Once you configure the first account and password successfully, the device may restart. After completing a new search, the lock icon will change to **Advance** type:



If there is an error during the unlocking process, like entering the wrong password, you will be notified with an error pop-up message such as "Unlock fail".

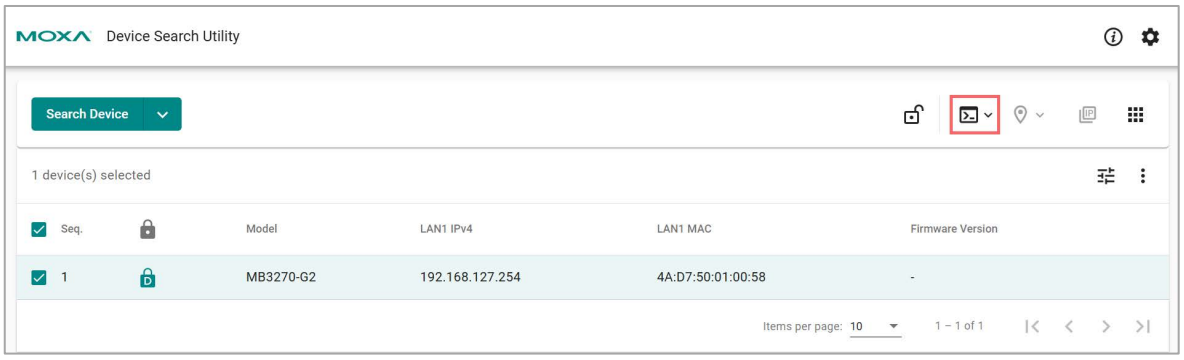
Unlock



When selecting one or multiple MGates, select the Unlock button to unlock them.

Console

When you want to configure more detailed settings, select the Console button to connect to the HTTPS console of the MGate MB3000-G2 Series.



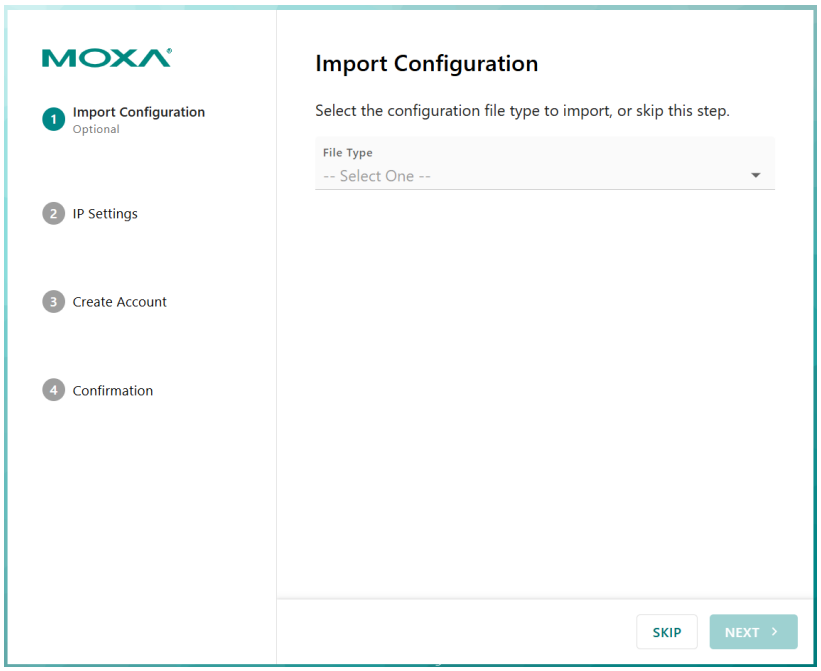
For more detailed features about DSU such as assign IP addresses, refer to the **Device Search Utility v3.x User Manual** from moxa.com.

First-time Login Process

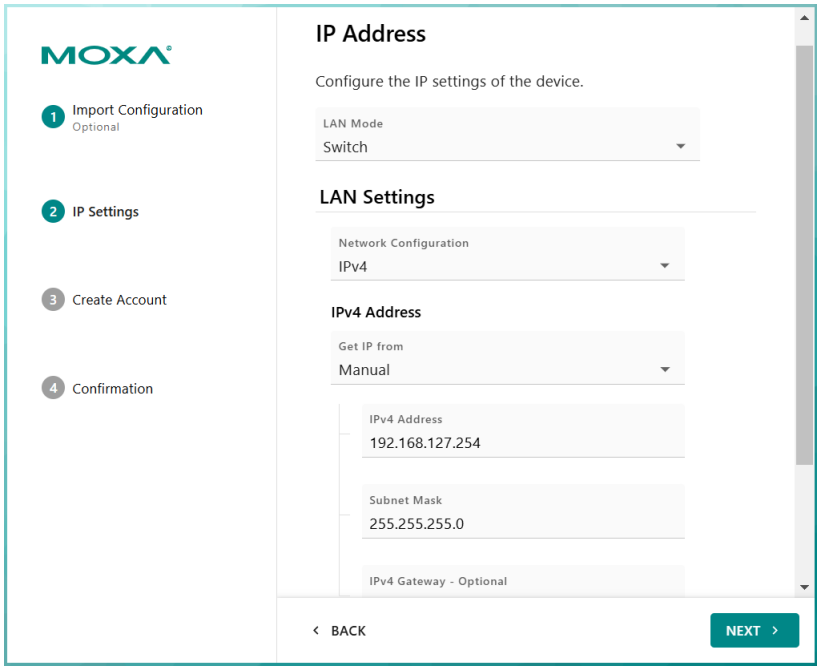
When you select the Console button at Device Search Utility or input the default IP address https://192.168.127.254 to log in for the first time to the web console of an MGate MB3000-G2 Series, a login wizard will guide you to initialize the device for setting up the first administrator and the network settings of this device. On the console webpage, select **START** to start the login process.

The first step is **Import Configuration**. If you have an existing configuration file from a MGate MB3000 or MB000-G2, select and import it in the first step. The MGate MB3000-G2 will then be configured with the settings from the old unit, and the wizard will proceed to the final step for your confirmation.

If you don't have an existing configuration file, select **SKIP** to skip this step.

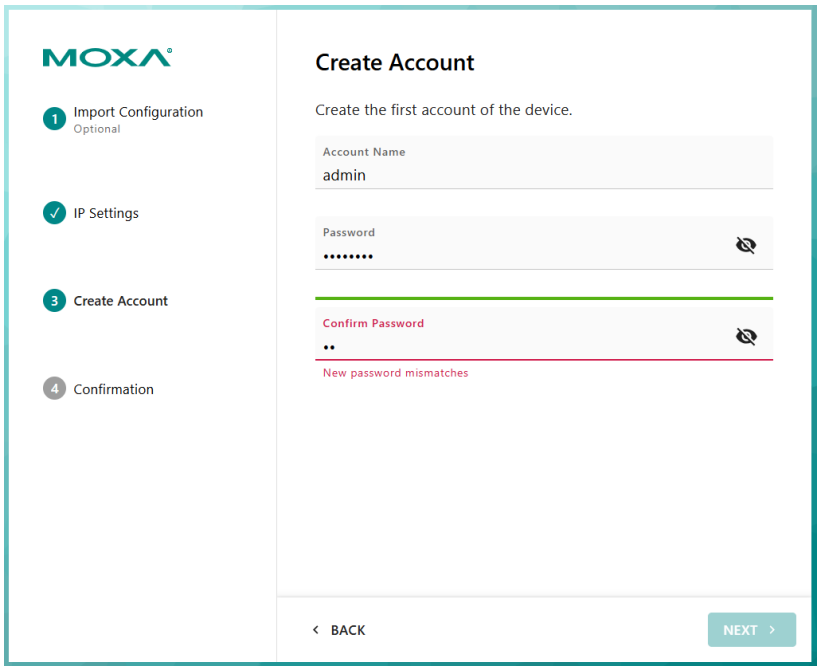


The second step is to check if you want to change the IP settings now. The default IP address is 192.168.127.254, and the netmask is 255.255.255.0. Modify LAN mode (**Switch** or **Dual IP**) or the IP settings to use DHCP or a different IP address based on your network topology. Select **NEXT** to proceed to the next step.

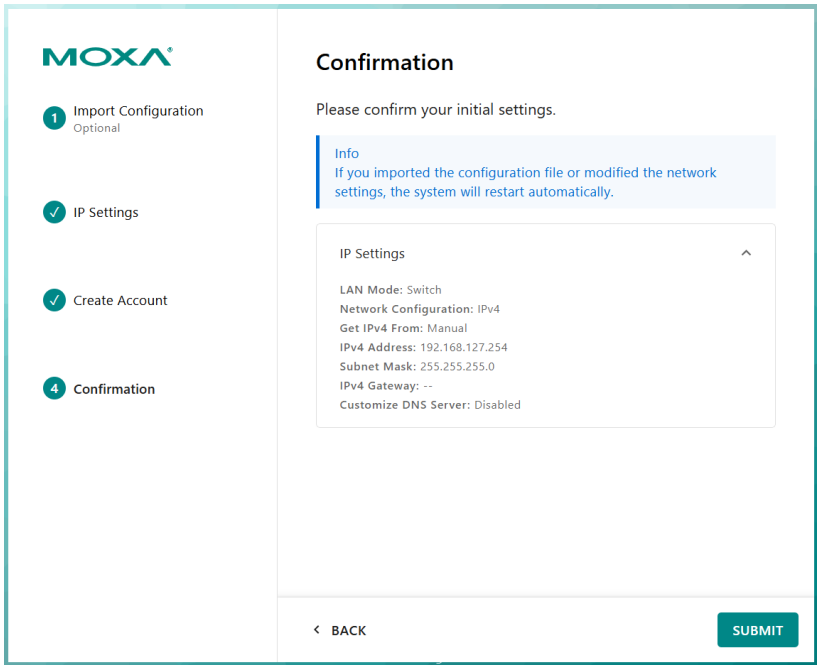


The third step involves creating a user account and password. Since there is no default username and password for MGate MB3000-G2 devices, you must create the initial account for this unit. This initial user account will have full privileges.

To enhance security, the default password must be at least eight characters long. You can further change password security rules by adjusting the Password Policy within the Account Management page.



Double-check the network settings at the "Confirmation" step. If everything is OK, select the **SUBMIT** button and the unit will reboot, applying the above settings.

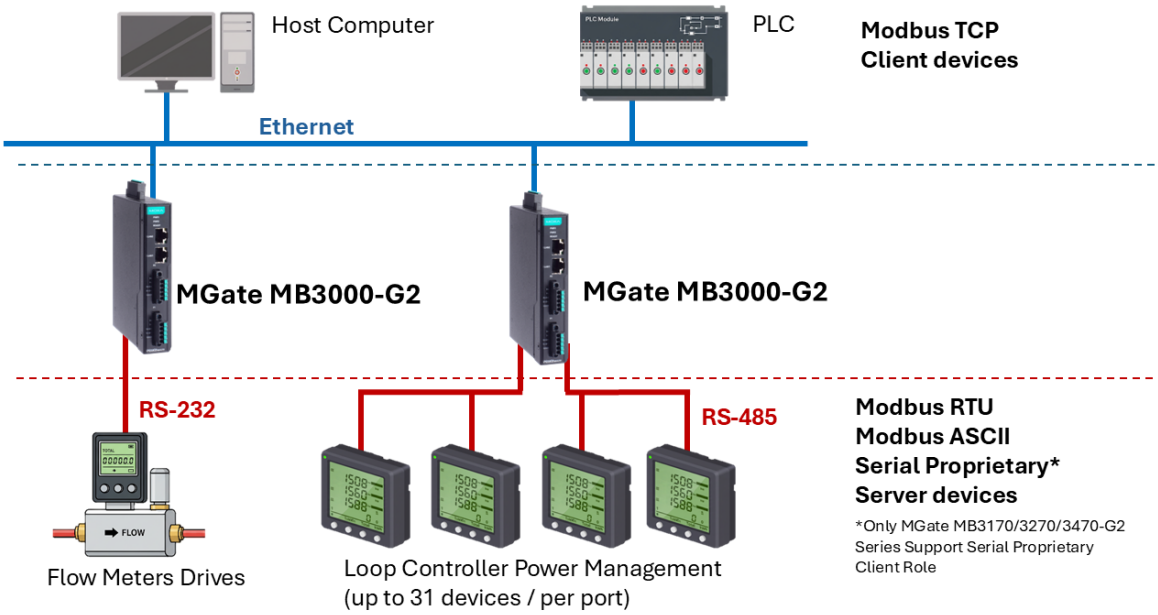


Modbus TCP Clients With Multiple Modbus Serial Servers

Connect all Modbus devices over an Ethernet network

Most modern PLCs and host computers support Modbus TCP that runs on Ethernet. To access Modbus RTU/ASCII devices for data collection and control, they can rely on the MGate MB3000-G2 Modbus gateway.

The MGate MB3000-G2 supports Modbus TCP with up to 16 or 32 simultaneous connections, depending on the model. The serial interface supports both RS-232 and RS-422/485, selectable through software. Each serial port can be connected to one RS-232 or RS-422 serial device, or up to 31 RS-485 serial devices.

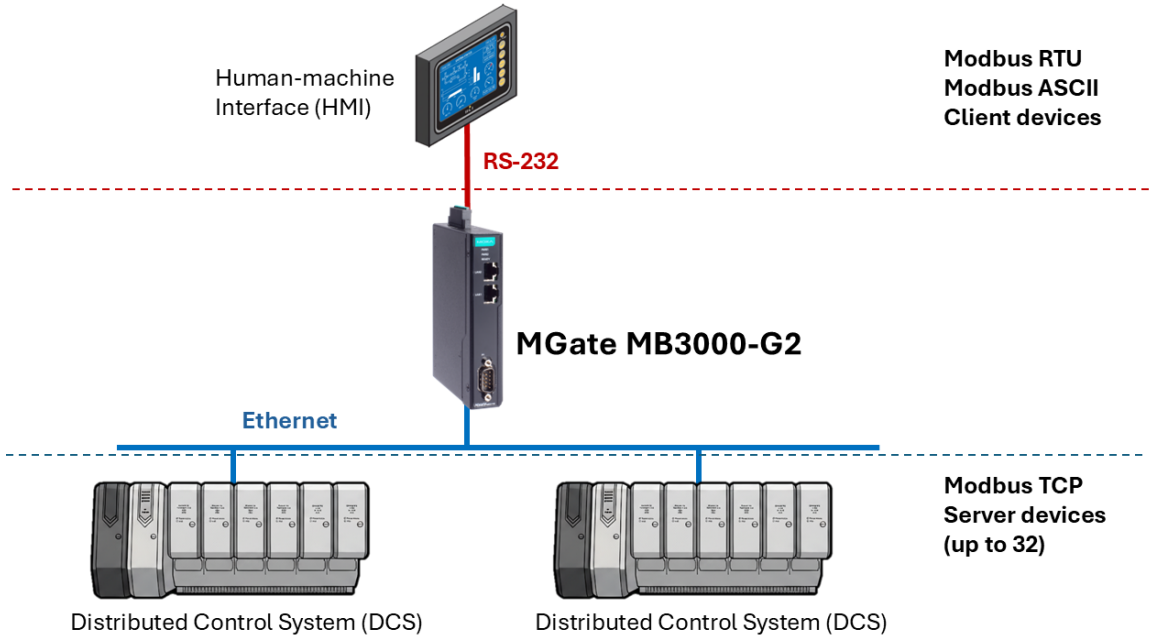


Modbus Serial Clients With Multiple Modbus TCP Servers

Link a Modbus serial device with Modbus TCP server devices

Many Human-machine Interface (HMI) systems use a serial interface to connect to a discrete Data Control System (DCS). However, many DCSs are now Ethernet-based and operate as a Modbus TCP server device.

The MGate MB3000-G2 Modbus gateway can link a serial-based HMI to distributed DCSs over an Ethernet network. Up to 32 Modbus TCP server devices are supported by each MGate MB3000-G2.

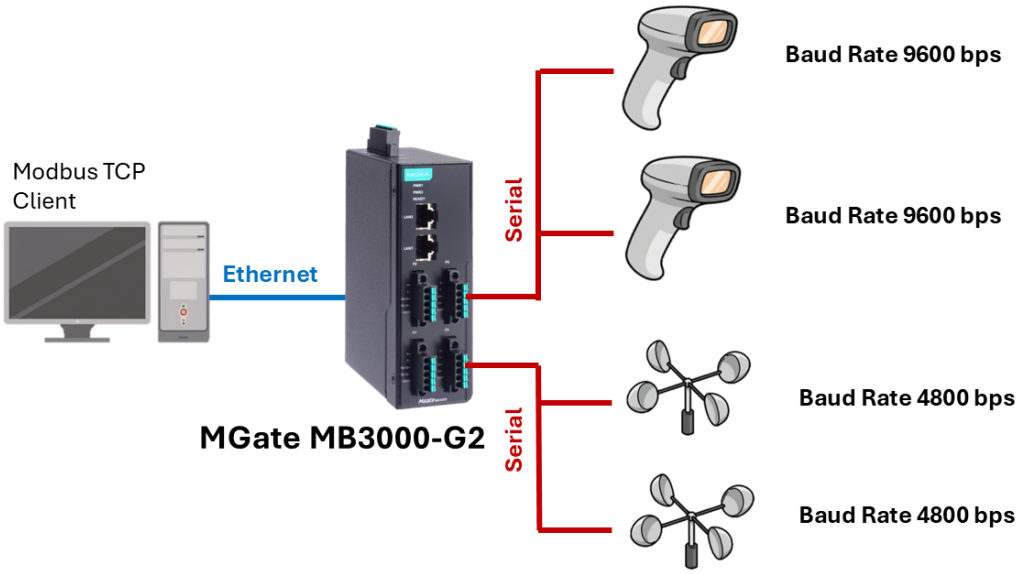


Modbus TCP Clients With Modbus ASCII and RTU Servers

Link TCP client devices with both ASCII and RTU serial devices simultaneously

When integrating Modbus networks, you may encounter different Modbus serial networks that use different baudrates or a different protocol. Modbus ASCII might be used by some devices, while Modbus RTU is used by other devices.

The two- and four-port MGate models can integrate serial Modbus networks that use different parameters or protocols. Configure each serial port to a specific Modbus serial environment; set up a device ID table. After configuration, only the gateway will be visible to Modbus TCP Clients, and all serial devices will be integrated behind it.

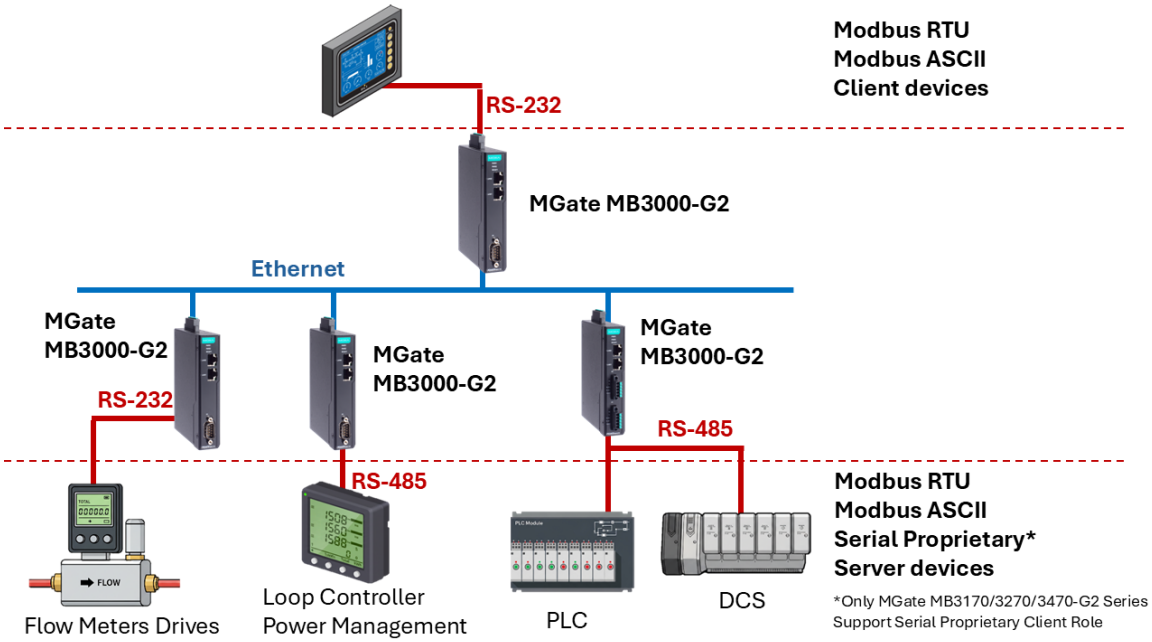


Modbus Serial Client With Modbus Serial Servers Over the Internet

Let Modbus serial devices communicate over the Internet

Many Modbus devices communicate over RS-485, which limits the number of devices in a network to 32 and the transmission distance to 1.2 km.

With the MGate MB3000 Modbus gateway, you can link all Modbus devices over an Ethernet network. Up to 32 Modbus gateways can be installed in a single control network, so each device can now be accessed from anywhere the TCP/IP network reaches.



5. Modbus Gateway Configuration

A Modbus gateway facilitates communication by converting between Modbus TCP and Modbus RTU/ASCII protocols. The most common application is to enable Modbus RTU end devices to communicate with Modbus TCP clients, such as SCADAs. This chapter provides instructions on configuring the MGate for Modbus protocol conversions.

Advanced Modbus gateway models, such as the MGate MB3170-G2, MB3270-G2, and MB3470-G2, support two operation modes:

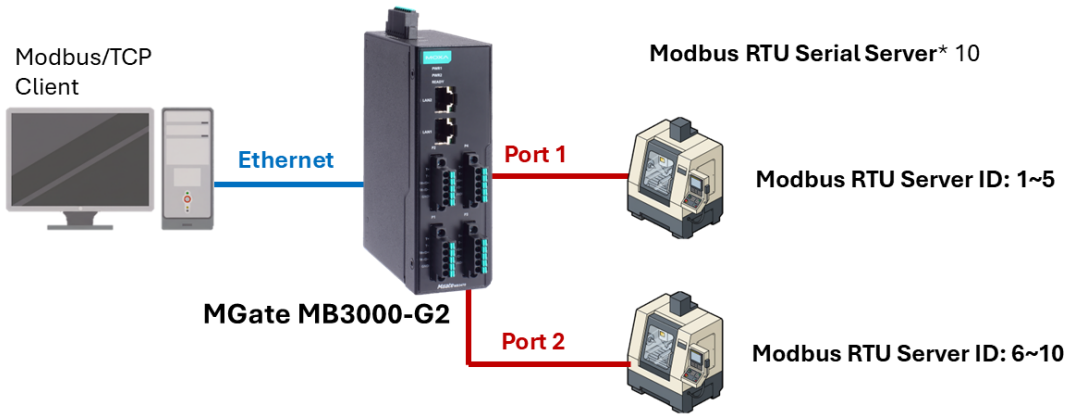
- **Transparent Mode:** The gateway acts as a transparent translator between Modbus TCP and RTU/ASCII.
- **Agent Mode:** The MGate actively polls Modbus devices and caches the data in its memory for enhanced polling performance.

Configuring MGate to Convert Modbus RTU Server End Devices to Modbus TCP Client Systems

Install Device Search Utility (DSU) v3.0. Select the **Search** icon to search for MGates.

Once the MGate's IP address is identified, enter it into the address bar of your web browser to access the web console.

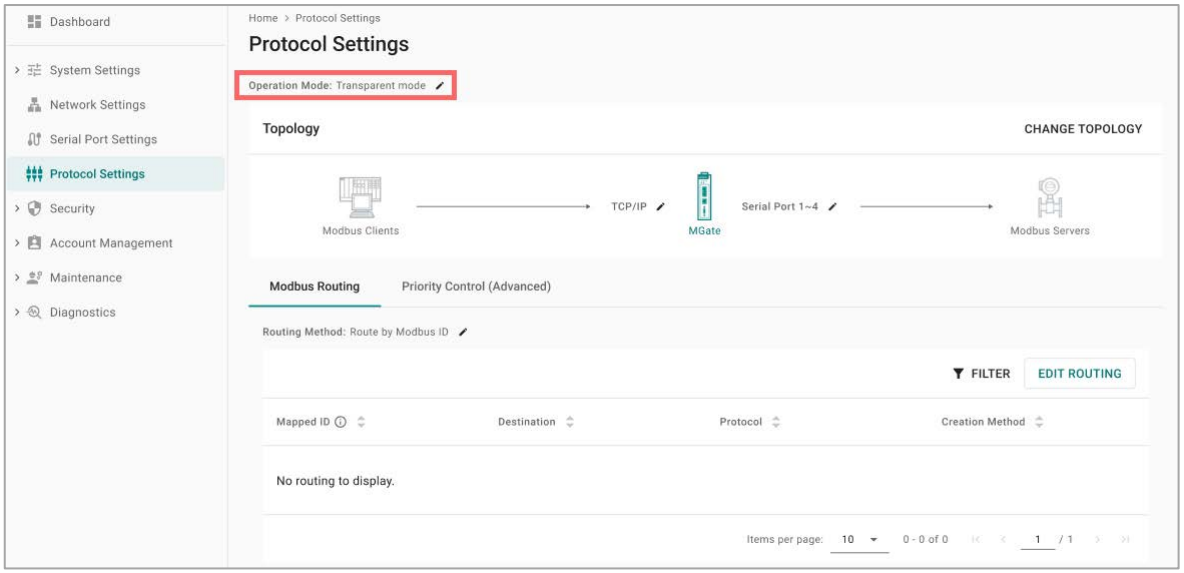
The following diagram illustrates the most common system design scenario requiring Modbus conversion. The two sections below will guide you through configuring Modbus conversion in transparent mode and agent mode.



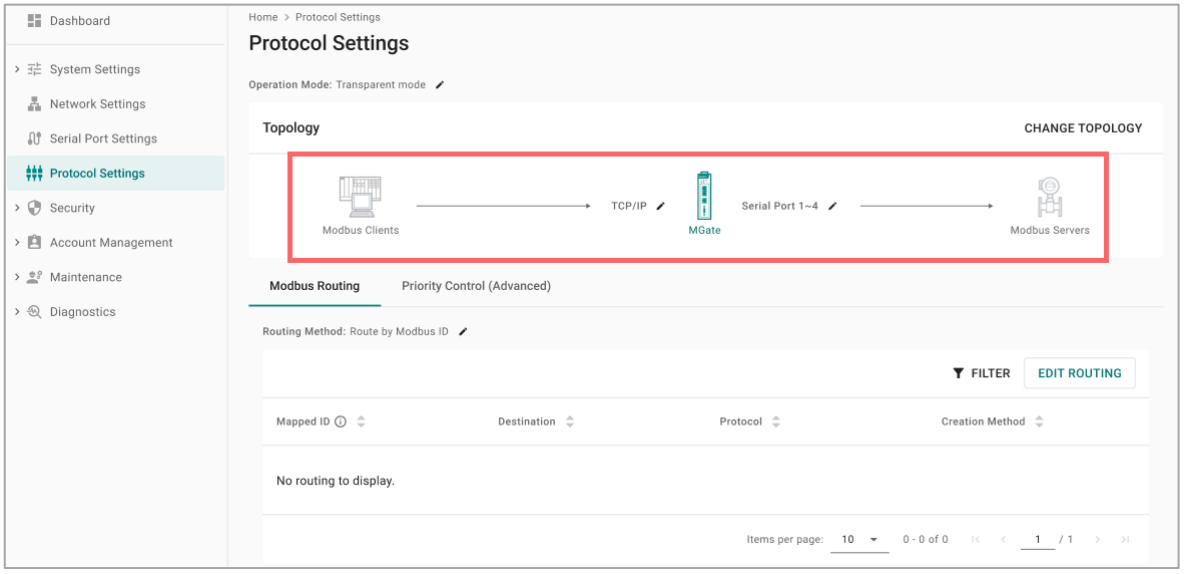
Configuring Modbus Conversion in Transparent Mode

To ensure successful Modbus conversion in transparent mode, check the following two settings: **Topology** and **Modbus Routing**. Details are provided below.

Go to the Protocol Settings page. The default operation mode is transparent mode, so you do not need to make any changes.



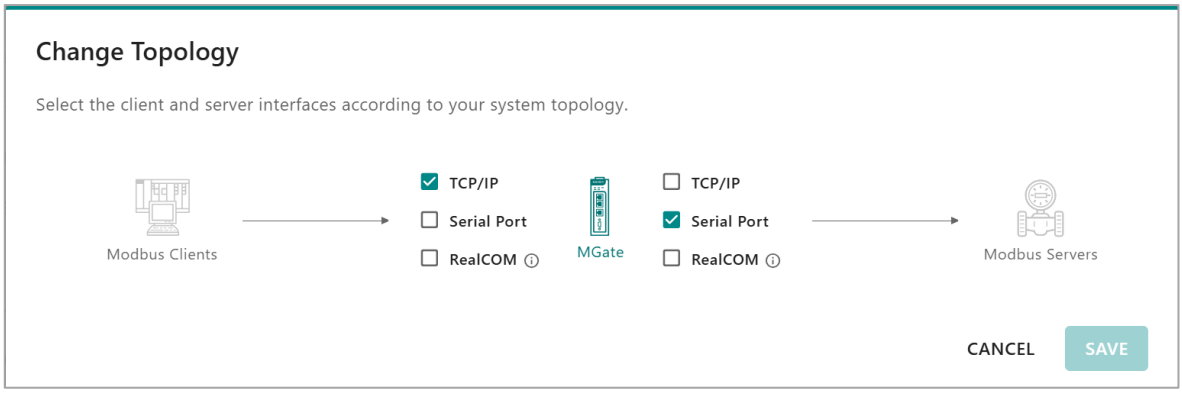
Check the Topology settings. The default topology, which converts Modbus RTU servers to Modbus TCP clients, is the most common configuration.



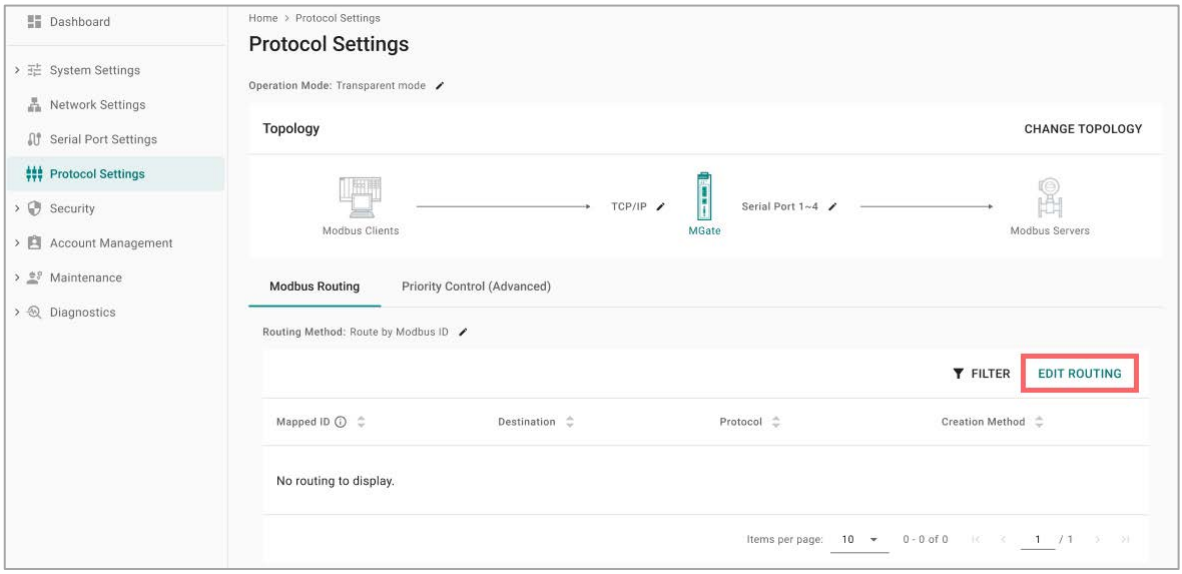
If you are using different or more complex architecture, select **CHANGE TOPOLOGY** and choose the desired system topology.

NOTE

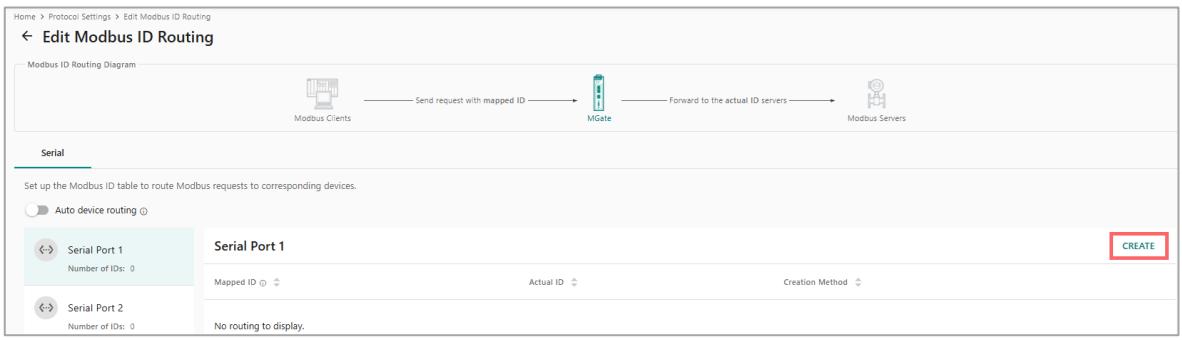
Only multiport MGate models (MGate MB3270-G2, MB3470-G2) with more than two serial ports can support the Modbus Serial-to-Modbus Serial topology. If you need to use the Real COM to simulate a virtual COM port on your PC, refer to the Real COM mapping section.



The final step is to configure Modbus Routing to ensure that Modbus IDs are forwarded to the correct serial port. Select **EDIT ROUTING** to start configuring the Modbus ID settings.

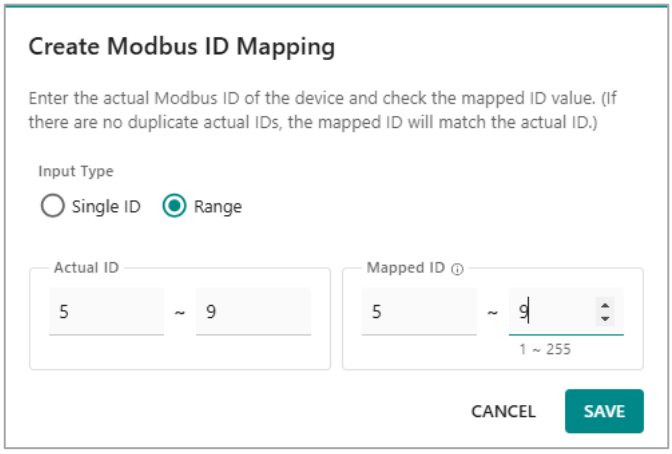


Select "CREATE" to define Modbus routing and how Modbus IDs are managed.



Configure a single ID or a range of IDs. In this example, since the serial port connects to a series of continuous IDs, we will use a range.

Enter the **Actual ID**, which represents the Modbus ID of the actual end device. Also, enter the **Mapped ID**, which represents the IDs that will be read by the Modbus TCP client. In the example, ID mapping is not required, and the Mapped ID will be identical to the Actual ID.

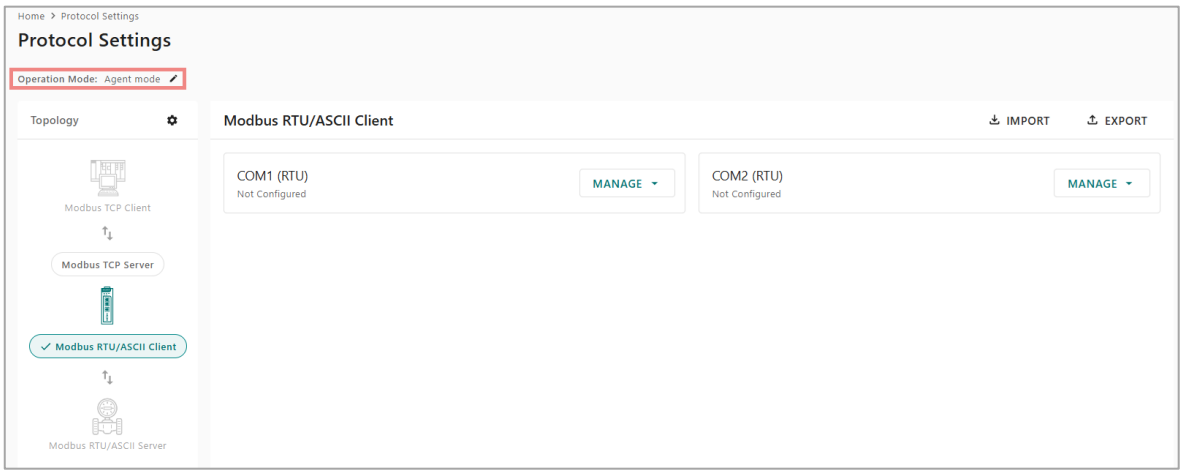


Once the Modbus routing settings are configured, you can begin to test communication.

Configuring Modbus Conversion in Agent Mode

To ensure successful Modbus conversion in agent mode, check the following two settings: **Operation Mode**, the **Devices** and **Commands** you'd like to poll, and the **Data Mapping** from end devices to the northbound protocol. Details are provided below.

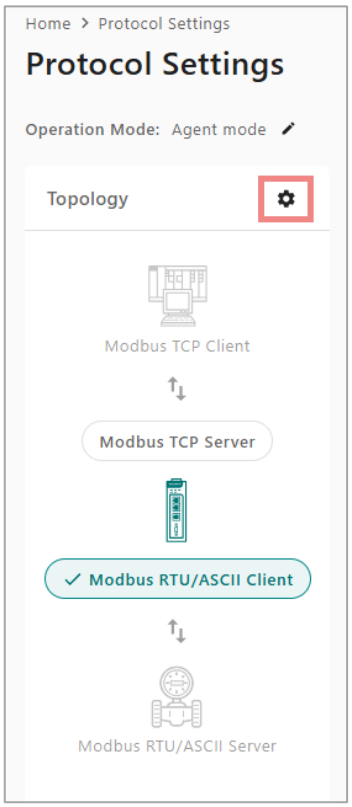
Go to the Protocol Settings page. Select the edit icon besides operation mode and change it to Agent Mode. After changing to Agent Mode, you will see the following.

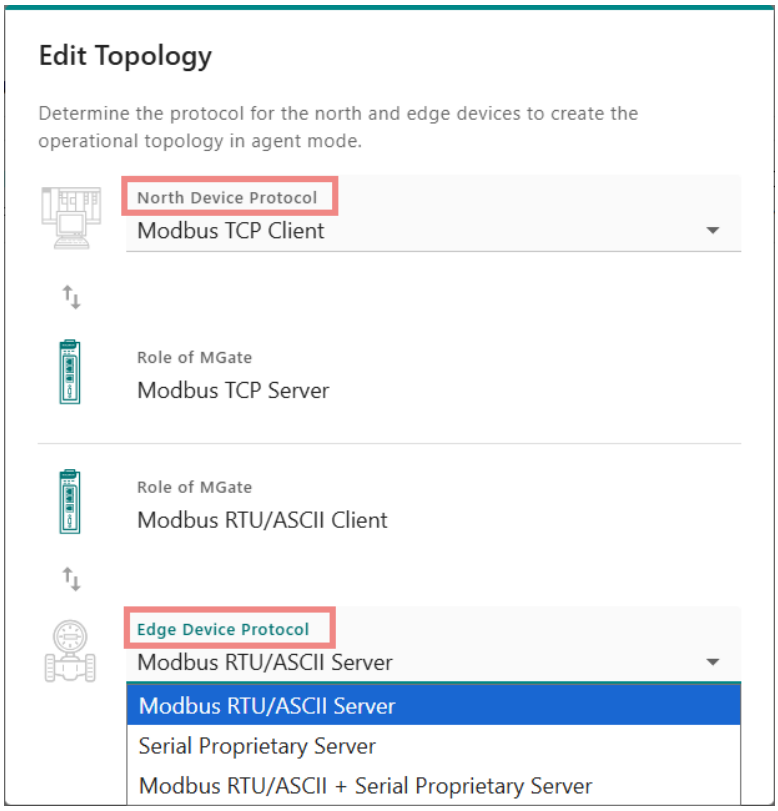


If you are using different or more complex architecture, select the icon beside Topology and choose the desired system topology.

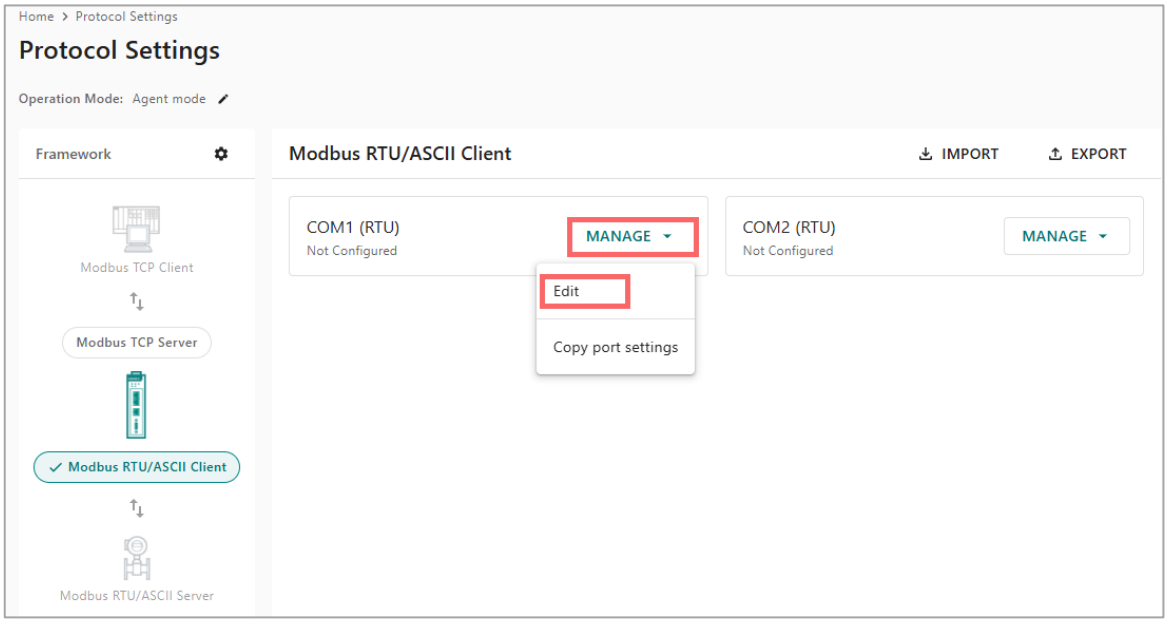
NOTE

Agent Mode typically supports Modbus serial/Serial proprietary-to-Modbus TCP or vice versa, only multiport MGate models (MGate MB3270-G2, MB3470-G2) can simultaneously support Modbus serial and Serial Proprietary-to-Modbus TCP topology.

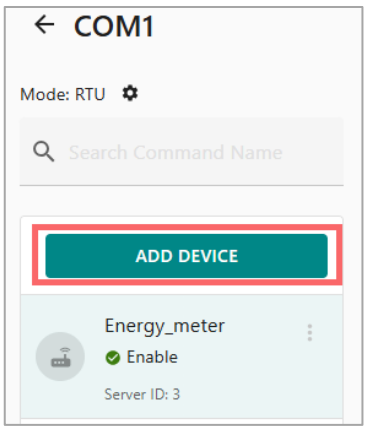




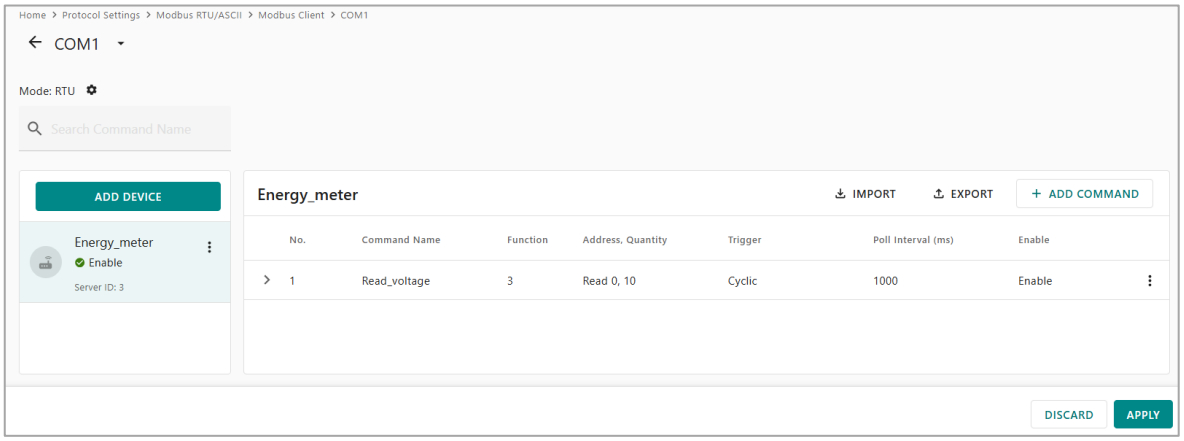
First we're going to configure the Modbus RTU/ASCII commands that MGate will send to end devices. Check if the topology matches your application. Under MGate's Modbus RTU/ASCII client role, select the desired serial port, and select **MANAGE > Edit**.



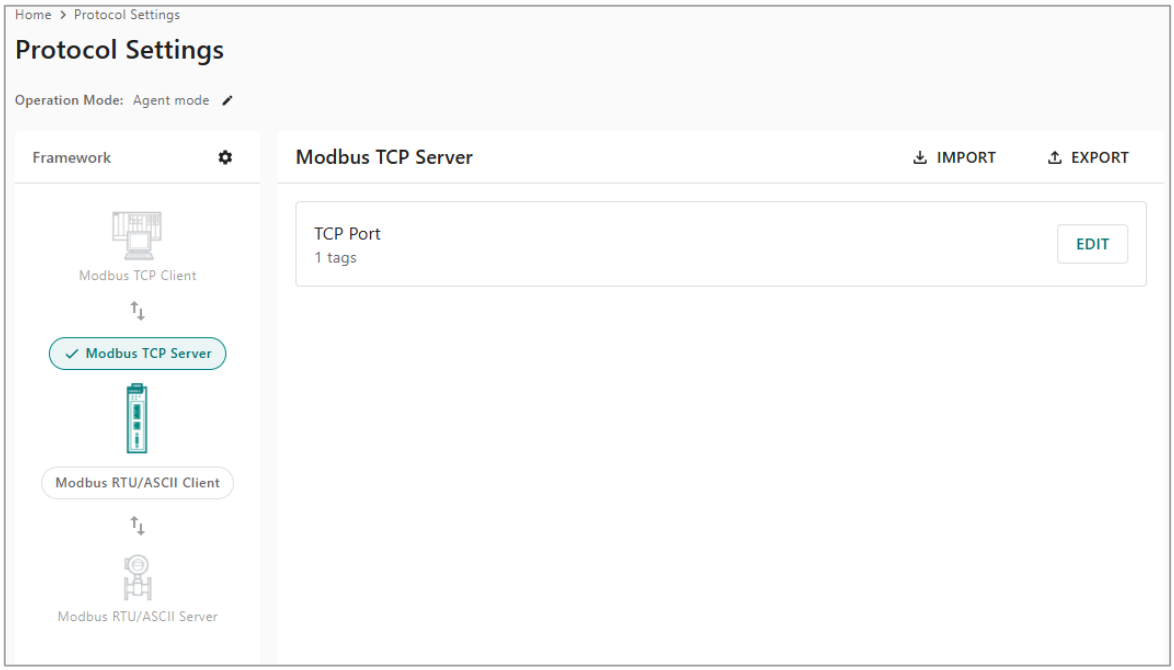
Select **ADD DEVICE** to add a Modbus RTU/ASII device that is connected to the MGate's serial port. For example, we create a device called the "Energy_meter". Then, set up the Modbus server ID, and the commands that the MGate will actively poll to Modbus servers. Here we create a command called "Read_voltage" and set the Modbus ID to be 1. After configuring commands, the MGate will create "Tags" of these data, which can be mapped to Modbus TCP later.



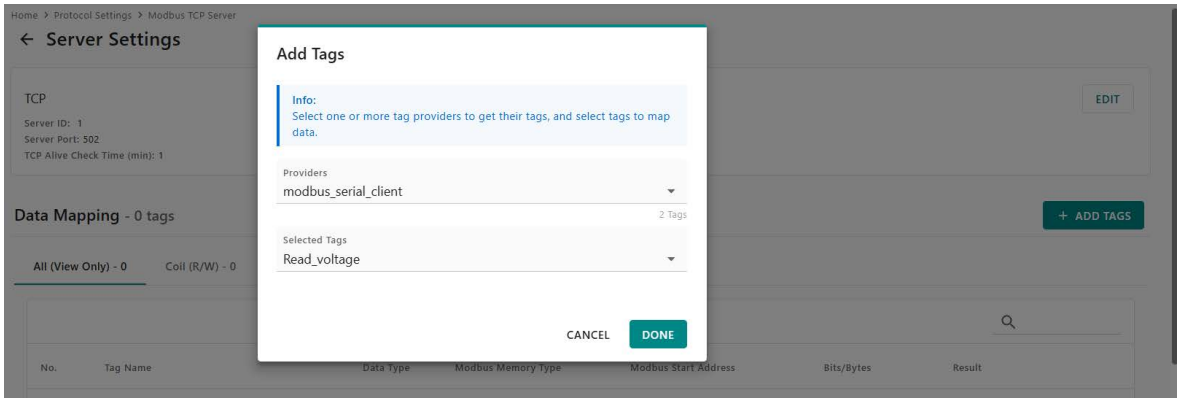
For these settings to take effect, remember to select **APPLY**.



Go back to the topology. Select the Modbus TCP server of the MGate to configure protocol mapping from Modbus RTU to Modbus TCP.



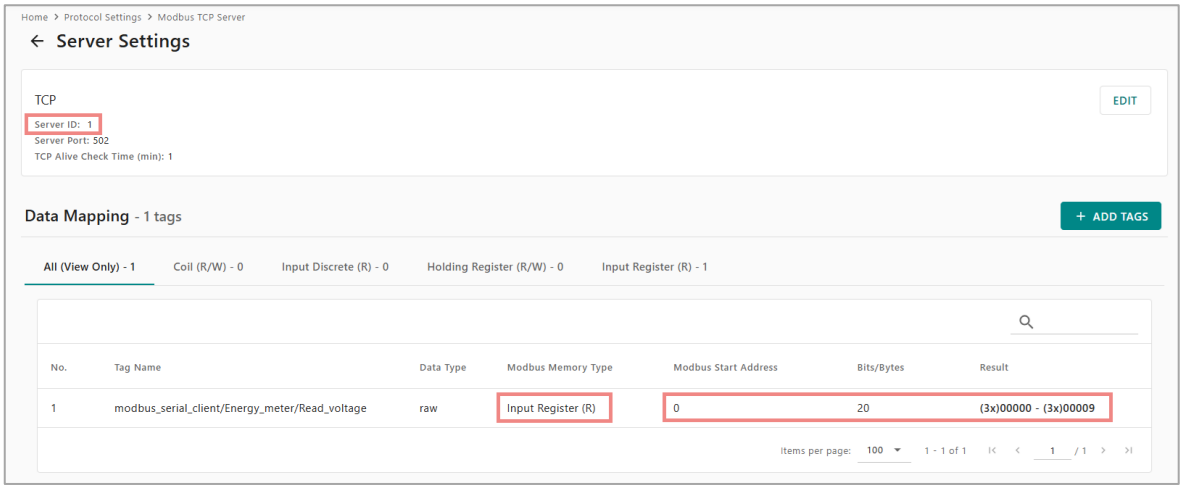
Select EDIT > ADD TAGS, to add the Modbus RTU tags created in the previous step. Here, select "modbus_serial_client" as the provider, and then the "Read_voltage" tag. Select DONE for the MGate to do the mapping.



The selected tags will display in the data mapping column by default with register/coil address. View all the data, or view by Modbus functions—Read/Write Coil, Read Discrete Input, Read/Write Holding Register, Read Input Register. You may adjust it manually. After making changes, select **APPLY** to take effect.

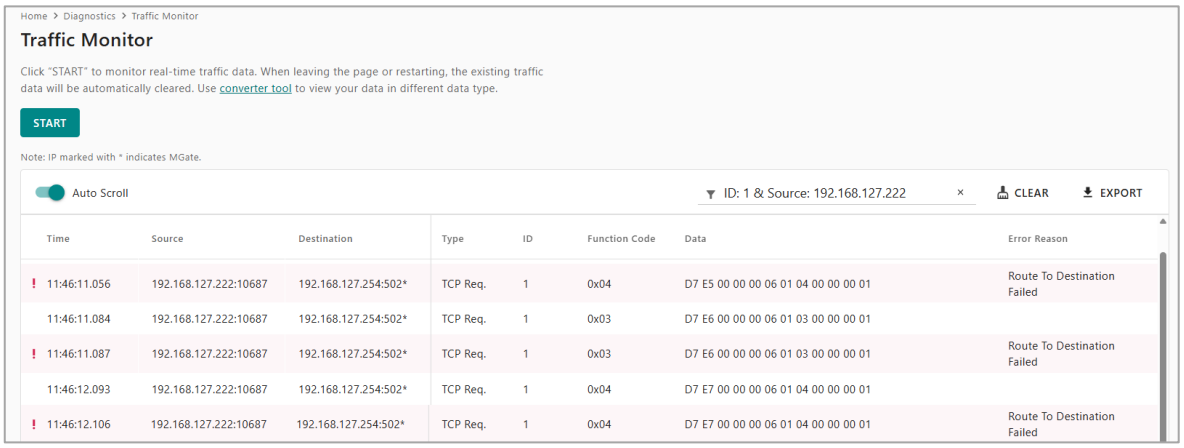
Now the protocol conversion settings are configured. Begin testing communication.

On the **Modbus TCP Server > Data Mapping** page, you can see the mapping information: Modbus TCP server ID 1, Input Register (R) means Modbus function code 04 (Read Input Register), and address from 0 to 9 which is 10 registers and total 20 bytes. Set up your Modbus TCP client to request this tag, and then you will get the end device's data.



Monitoring Modbus Activity

For troubleshooting or management purposes, you can monitor the data passing through any MGate MB3000-G2 on the network. You can monitor the Modbus traffic passing through the MGate (Transparent Mode) or the traffic between Modbus Client (master)/Server (slave) and MGate (Agent Mode). The MGate Traffic Monitor presents the data in an easily understood format, with clearly designated fields including timestamp, source, destination, types of Modbus frame, Modbus ID, Function Code, Data. and error reason. Filter events by source, destination, ID, Function Code, and error reason, and save the complete log to a file for later analysis. For more details, check **Diagnostics > Traffic Monitor**.



Network Management Tool (MXstudio)

For the software and related detailed information regarding MXview One and MXconfig, as well as the supported product firmware versions, refer to the Moxa website at https://www.moxa.com/en/products/industrial-network-infrastructure/network-management-software.

When you discover a Moxa product that has not been integrated into MXview One or MXconfig, you may not be able to retrieve the product information from MXview One or MXconfig. To solve this, you can download the plugin file from the Moxa MGate product website and then import/install the plugin into MXview One or MXconfig. After importing/installing the plugin files, the MGate products can be supported by MXview One/MXconfig. Refer to the Moxa MGate product website to download plugin files: http://www.moxa.com. For more detailed functions such as supported functions on MXview One/MXconfig, refer to the **Tech Note: Configuring and Monitoring with MXview One/MXview and MXconfig**.

6. Case Studies

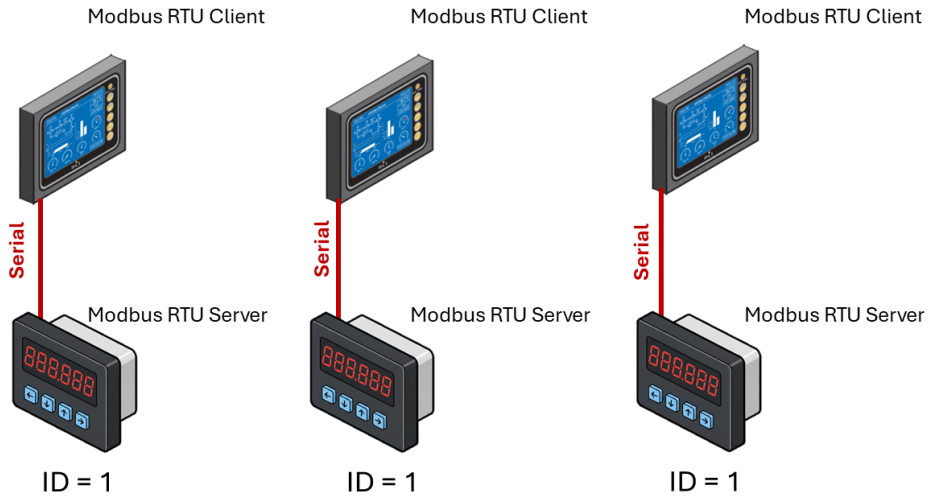
Introduction

There are many reasons a Modbus gateway might be used to integrate Modbus networks. However, every situation has its own requirements and difficulties. You may wonder how the gateway can help, or even if the gateway is suitable for the system.

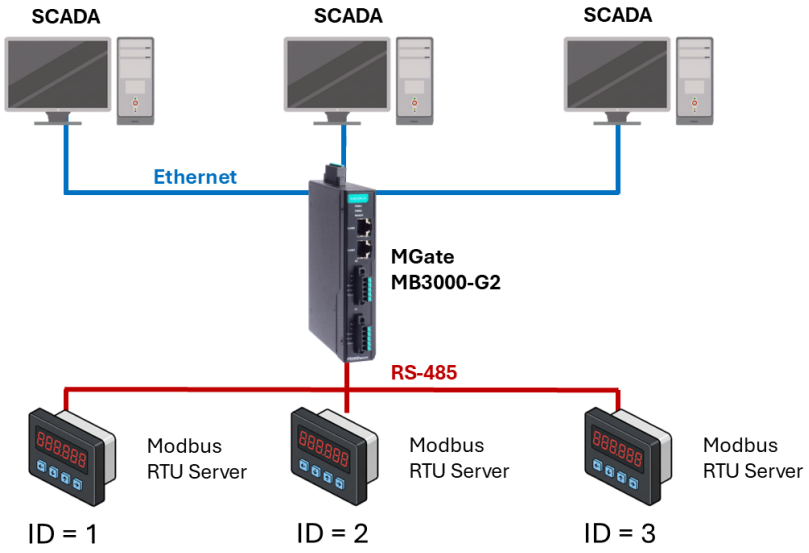
This chapter presents some case studies to guide you. If you cannot find a case like yours, it does not mean the MGate MB3000-G2 is not suitable for you. Contact Moxa, and we will work it out with you.

Replace Modbus Serial Clients With Modbus TCP Clients; Modbus IDs Configurable

In this scenario, the original control system comprises several serial-based systems. In each system, a serial client directly controls serial server devices, as follows:

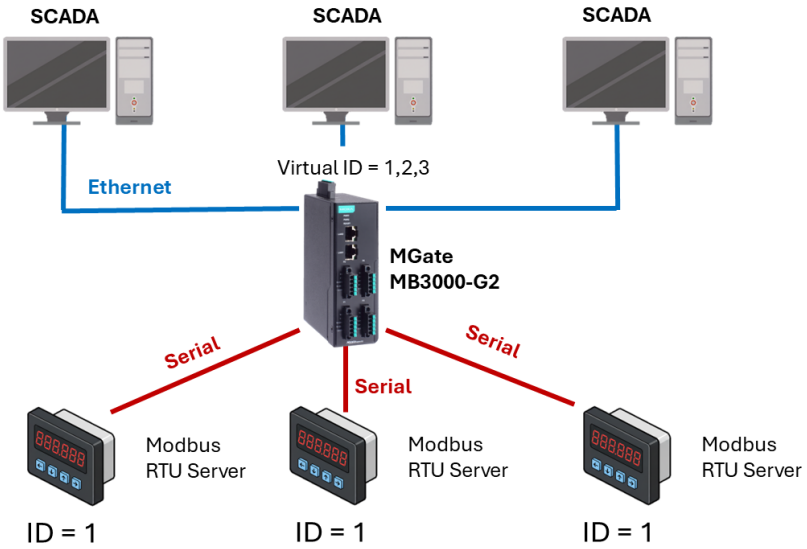


The MGate MB3000-G2 can connect to each multiple Modbus serial server devices, so Modbus TCP SCADA clients will be able to monitor or control them. However, since Modbus IDs cannot be repeated in a system, we will need to change the Modbus IDs of some servers to integrate them into a single network, as follows:



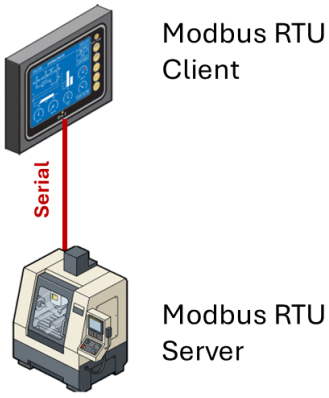
Replace Modbus Serial Clients With Modbus TCP Clients; Modbus IDs Are Fixed

Some legacy Modbus server devices have fixed IDs that cannot be changed. To integrate the devices into a Modbus TCP network, a multiport MGate model (MGate MB3270-G2 or MGate MB3470-G2) supports Mapped ID function to change the requests' IDs from the Modbus TCP client, so repeated Modbus IDs can still be used in the system. For details, refer to the Protocol settings section below.



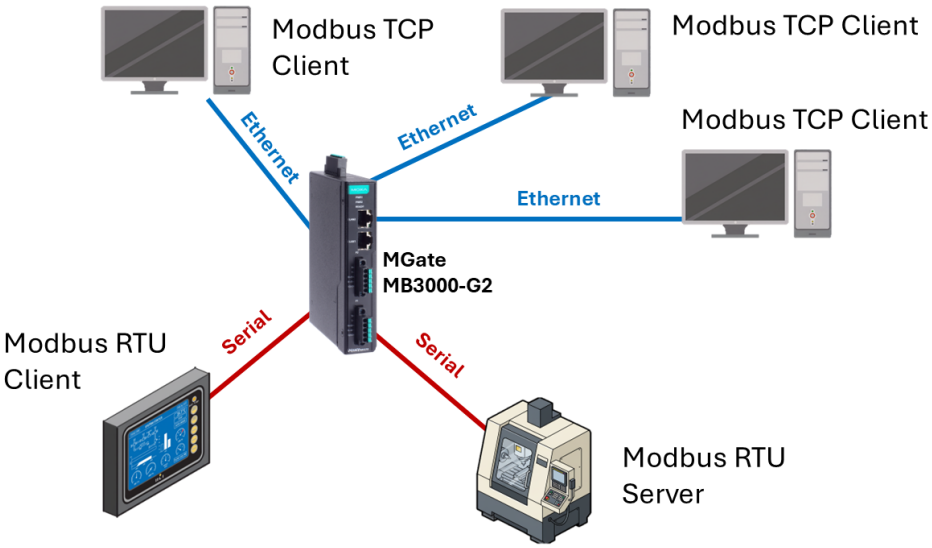
Keep Modbus Serial Clients and Add New Modbus TCP Clients

In this scenario, the serial control system is a direct, low-latency system. The Modbus serial client keeps operating with the Modbus serial servers, but the system needs to add new Modbus TCP clients to also monitor the Modbus serial server end devices for monitoring or supervision.



Serial Redirector: Modbus Serial to Modbus Serial

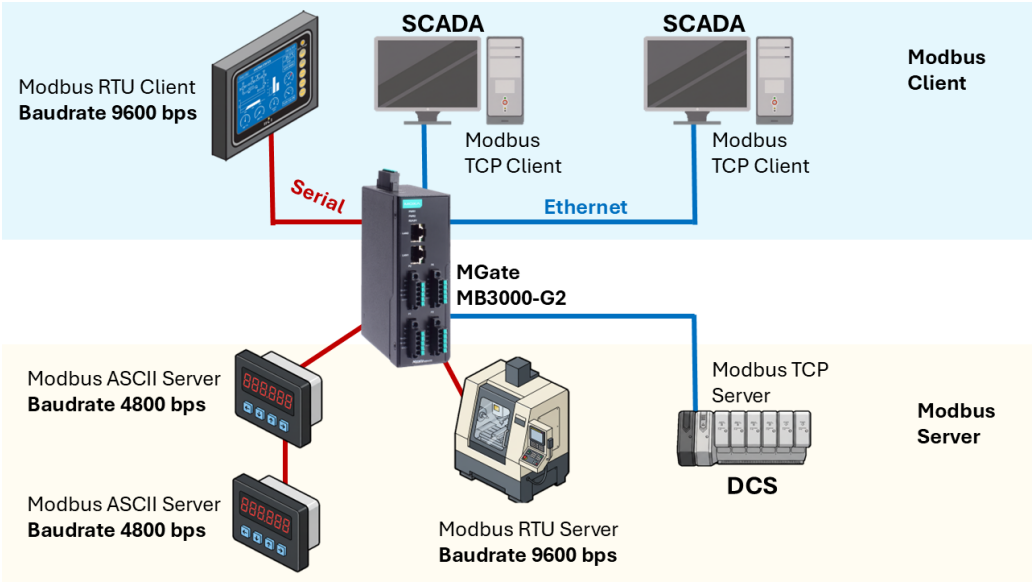
An advanced multiport MGate model (MGate MB3270-G2 or MB3470-G2) supports the serial redirector function, which is able to integrate a Modbus serial to Modbus serial system, and a Modbus serial to Modbus TCP system simultaneously. Many Modbus serial control systems in the field and local control devices, such as HMI, are connected to Modbus serial field devices. Using Ethernet-based equipment for remote access and monitoring has become a trend. By setting up the MGate, you will keep the original Modbus serial control system and add Modbus TCP client (e.g., SCADA) or/and Modbus TCP server (e.g., PLC) into the system. Both Modbus TCP and Modbus RTU/ASCII clients can control Modbus TCP and Modbus RTU/ASCII servers. The MGate can act as a "Serial Redirector" by configuring the Protocol Settings. For more information about how to set up the serial redirector function, refer to the Protocol Settings in this manual.



Integrate Modbus RTU, ASCII, and TCP at the Same Time

There can be a wide range in number, type, and sophistication of devices on the factory floor. The most common devices are simple Modbus serial-based meters, which report certain information relating to factory environment or equipment. However, other Modbus serial servers may be as complex as a manufacturing machine or a PLC controller.

When integrating these devices, there may be issues if different serial environments are used for different devices. One system may use a different baudrate than another or may use Modbus ASCII instead of Modbus RTU. The MGate MB3000-G2 allows the different Modbus protocols (TCP/RTU/ASCII) with different baudrates to be integrated into one network, providing flexibility for users integrating a complex system.



7. Cybersecurity Best Practices

As cyberattacks become increasingly sophisticated, network device vendors are incorporating features to safeguard sensitive information. Moxa prioritizes developing measures to ensure all products meet stringent security standards, providing customers with peace of mind.

However, building a robust cybersecurity environment requires collaboration. Moxa and customers must work together to defend against the evolving landscape of cyber threats.

This chapter outlines essential steps to enhance the cybersecurity of Moxa products. For specific settings and commands, refer to other sections of this user manual.

For comprehensive cybersecurity guidance, refer to the **Security Hardening Guide for the MGate MB3000-G2 Series**.

Updating Firmware

Customers who buy products from Moxa or a reseller should be aware that Moxa might have already launched a newer firmware version with enhanced security features. Check Moxa's support website to see if there is a newer version of firmware. If so, we recommend upgrading the firmware to the newest.

Turn Off Unused Service and Ports

Imagine living in a house that has many entrances. If all the doors and windows are left unlocked or even open, it sends a message of welcoming to intruders out there. We always recommend turning off services and ports that are not in use to reduce the chances of being attacked. Refer to the table below for all the ports, protocols, and services that are provided to communicate between the MGate MB3000-G2 Series and other devices.

Service Name	Option	Default Settings	Type	Port Number	Description
Moxa Services	Enable/ Disable	Enable	TCP	443	For Moxa utility communication
			UDP	5353	
SNMP Agent	Enable/ Disable	Disable	UDP	161	SNMP handling routine
HTTPS Server	Enable/ Disable	Enable	TCP	443	Secured web console
DHCP Client	Enable/ Disable	Disable	UDP	68	The DHCP client needs to get the system IP address from the DHCP server
SNTP	Enable/ Disable	Disable	UDP	Random port	Synchronize the time settings with a time server
Remote System Log	Enable/ Disable	Disable	UDP	Random port	Send the event log to a remote log server
Real COM Mode	Enable/ Disable	Disable	TCP	950 to 953, 966 to 969	Moxa proprietary virtual COM communication mode.

Turn On Services That Are Necessary

Some services are recommended to be enabled because they are the key functions of the MGate MB3000-G2, and they face cybersecurity threats. The communication of these services is encrypted on the Ethernet network.

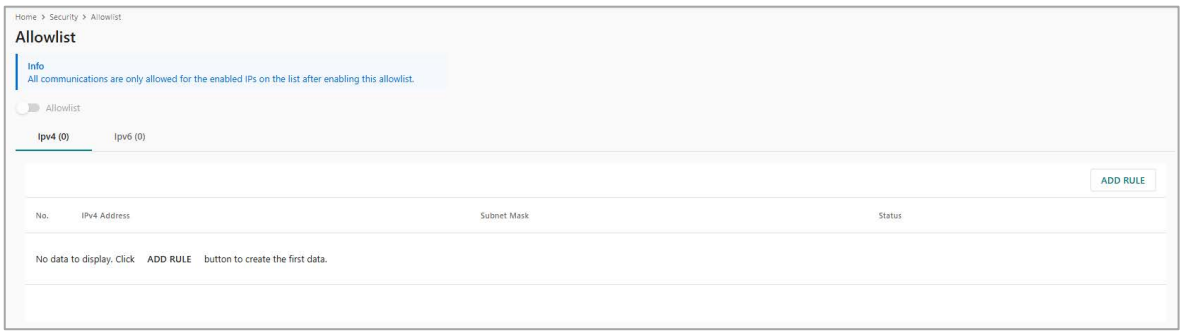
- Web console (HTTPS): This is the major management console of the MGate MB3000-G2 for configuring all the settings, and it also provides some diagnostic tools for an engineer to troubleshoot a problem.
- SNMPv3: The Simple Network Management Protocol is a popular tool for remote device monitoring and management. Enable SNMPv3 to encrypt communication data if needed.
- Moxa services: The Device Search Utility v3.0 is a good tool for first-time installation on the MGate MB3000-G2 Series, and Moxa MXview One and MXconfig can easily monitor all the MGates in a network. All these tools require the **Moxa services** to be enabled. For details about MXview One configuration, refer to the **Tech Note: Configuring and Monitoring with MXview One/MXview and MXconfig (non-java and java) Plugins**.
- Remote Syslog service: The system log is an important message for an engineer to analyze a problem. If the system has a central log server, the MGate MB3000-G2 supports syslog-ng to send the logs to the server securely.

NOTE

If all HTTP/HTTPS/Moxa services are turned off, then there is no other route to access the product. The only way to recover it is to reset the device and start from the beginning. For guidance on resetting the device, refer to the user manual.

Limited IP Access

Limiting the number of IP addresses that can access the product is one of the most effective ways of blocking unwanted intruders. If the product is accessed by a limited number of desktop/notebook/mobile devices, provide access to those IPs. The MGate MB3000-G2 has the Allowlist function to grant an IP address or a range of devices to access the device server. You can **ADD RULE** for those granted IP addresses and then enable the Allowlist function to limit access to the specific Gate MB3000-G2 only to those IP addresses.



Account and Password

There is no default username and password for MGate MB3000-G2 devices. You may need to follow up the first-time login process to set the username and password for the first user (who will also be the admin user) of this device to enhance the device's security.

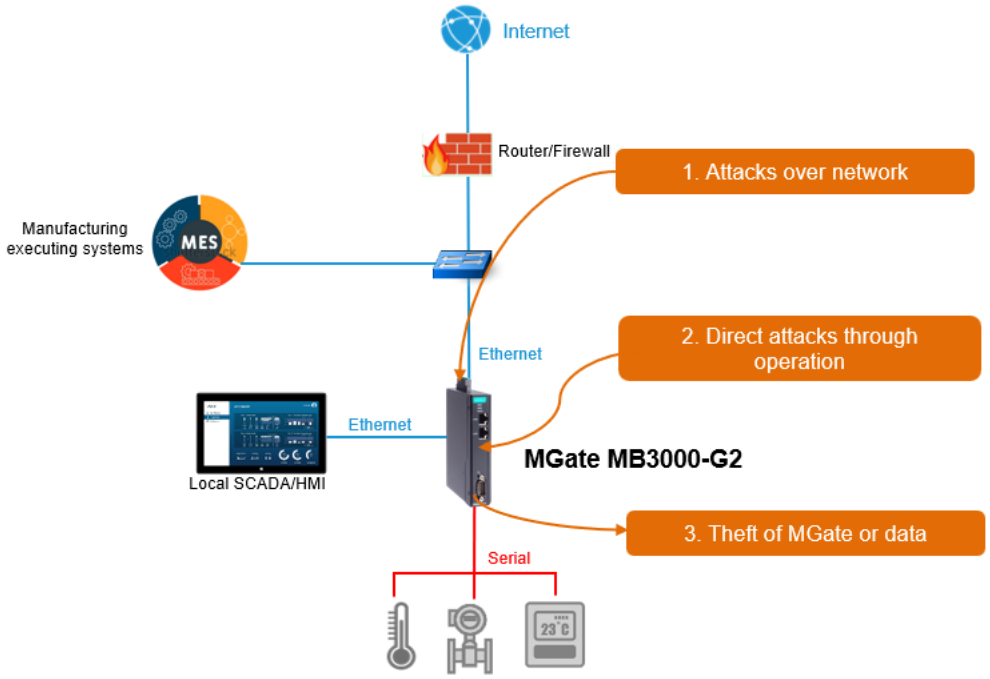
- Use strong passwords. The devices support a function called **Password Policy** to check if passwords are strong enough. Enable the function to help you check whether the passwords are strong enough.
- Use the account login failure lockout feature to prevent unwelcome access (**Security > Login Settings > Login Lockout**).

System Log

The system log usually records all kinds of activities that are happening on your MGate, such as Login Fail, IP Changed, Password Changed, Config Changed, etc. Check the log regularly to examine any abnormal behavior. For central management purposes, set up a log server in the network to collect all the logs from different devices. The MGate MB3000-G2 Series supports syslog-ng protocol to deliver the logs securely to the log server. The events will be sent with the format defined by RFC3164 for the analyzer to read/analyze. Refer to **System Settings > Notifications > Channels Settings** for more information.

Deployment of the Device

Deploy the MGate MB3000-G2 Series behind a secure firewall network that has sufficient security features in place to ensure that networks are safe from internal and external threats. Make sure that the physical protection of the MGate devices and/or the system meets the security needs of your application. Depending on the environment and the threat situation, the form of protection can vary significantly.



Testing the Security Environment

Besides these devices that support these protective functions, network managers can follow several recommendations to protect their network and devices. To prevent unauthorized access to a device, follow these recommendations:

- Testing tools for cybersecurity environment checks are available. Some may provide limited free use, for example, Nessus. These tools help identify probable security leaks in the environment.

  The device should be operated inside a secure network, protected by a firewall or router that blocks attacks via the Internet.
- Control any physical access to the device.
- Avoid using insecure services such as SNMPv1 or v2; the best way is to disable them completely.
- Limit the number of simultaneous web server sessions allowed. Periodically, change the passwords.
- Back up the configuration files periodically to make sure the devices work properly.
- Audit the devices periodically to make sure they comply with these recommendations and/or any internal security policies.
- If there is a need to return the unit to Moxa, make sure encryption is disabled, and that you had already backed up the current configuration before returning it.

NOTE

DISCLAIMER: The information and guide (the "information") in this section are for your reference only. We do not guarantee a cyberthreat-free environment. These guidelines are to increase the security level to defend against cyber intrusions and do not guarantee that the above information will meet your specific requirements. The above information is provided "as-is", and we make no warranties, express, implied or otherwise, regarding its accuracy, completeness, or performance.

8. Web Console Configuration and Troubleshooting

Factory Default IP Address

The MGate MB3000-G2 is configured with the following default private IP address: **192.168.127.254**.

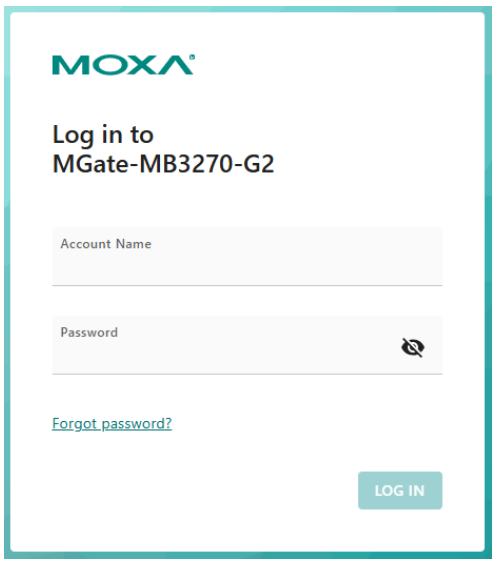
NOTE

IP addresses that begin with "192.168" are referred to as private IP addresses. Devices configured with a private IP address are not directly accessible from a public network. For example, you cannot ping a device with a private IP address from an outside Internet connection.

Using Your Web Browser

Opening the Web Console

Open your web browser, use HTTPS, and enter the IP address you've changed in the website address line. Press **ENTER** to load the page.

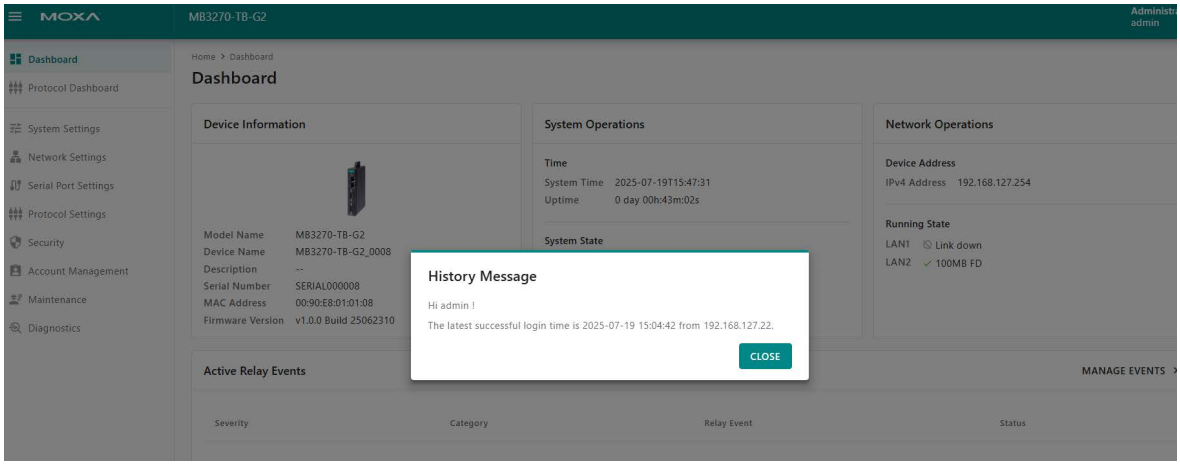


You may find the "Not secure" icon on the browser's website address line. Select it to add the MGate as a trusted device to remove the icon. For more information, refer to the tech note - **Security Hardening Guide**. Enter the account name and password you've set to access the MGate device.

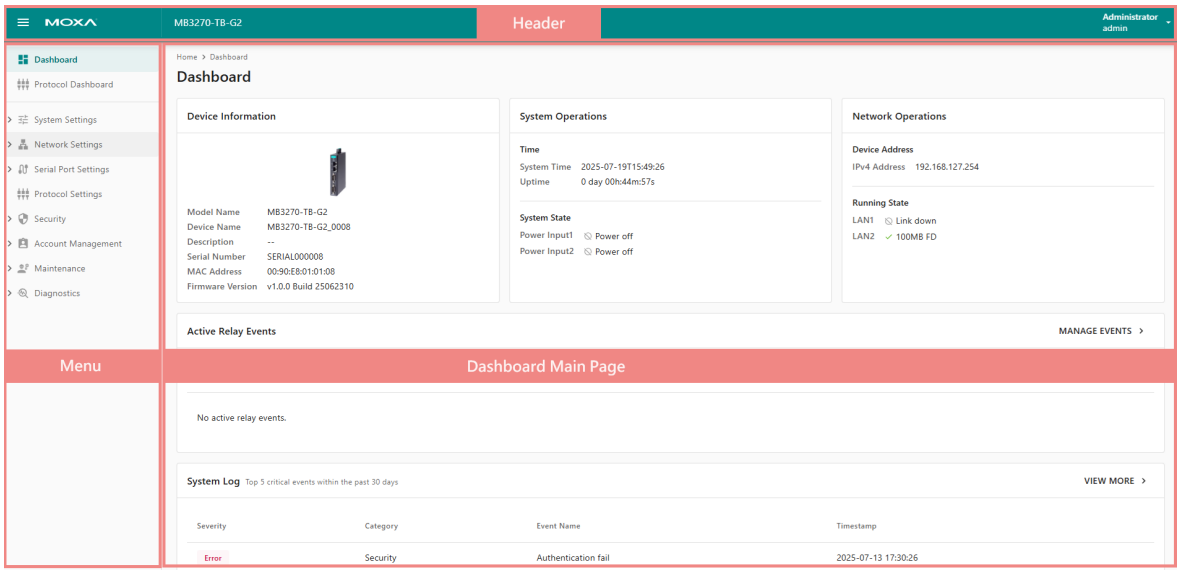
ATTENTION

If you forget your password, use the reset button to reset all MGate settings to factory defaults. Even if you disabled the reset button, you could still use it within the first minute of powering on the device to restore factory defaults. For easy maintenance, back up your configuration by exporting it to a file before using the reset button.

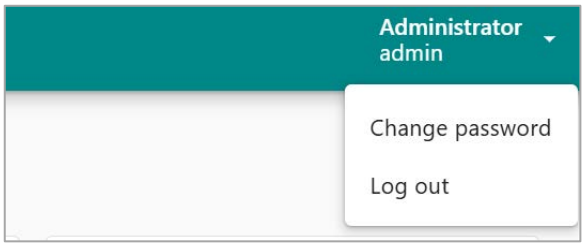
After logging in, the MGate MB3000-G2's web console will appear, displaying history messages, including the **Login Message** (configurable at **Security > Login Settings > Login Message**) and account login history.



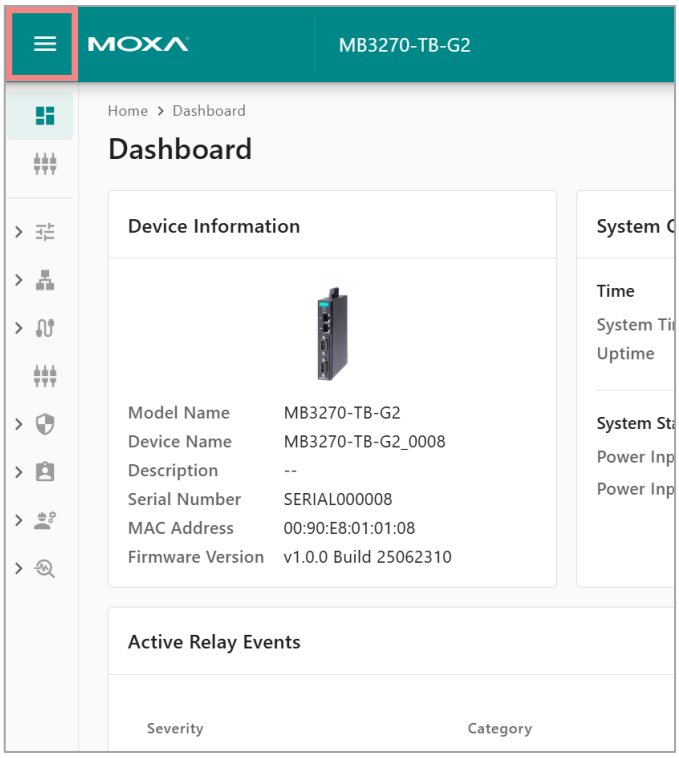
Select the **CLOSE** button and the Dashboard page will be displayed.

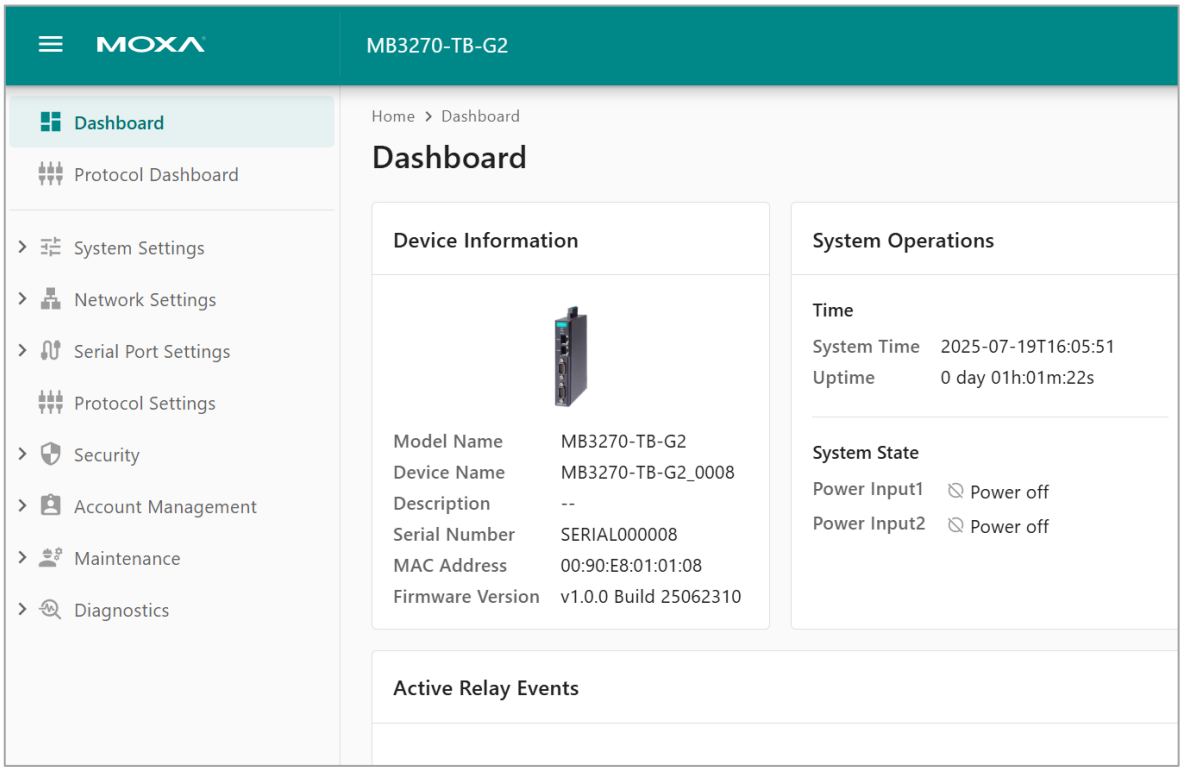


The Header shows who is logged in to the device. Select the account to change your password or log out the web console.



Select the icon on the upper left side to hide or show the Navigation Panel.





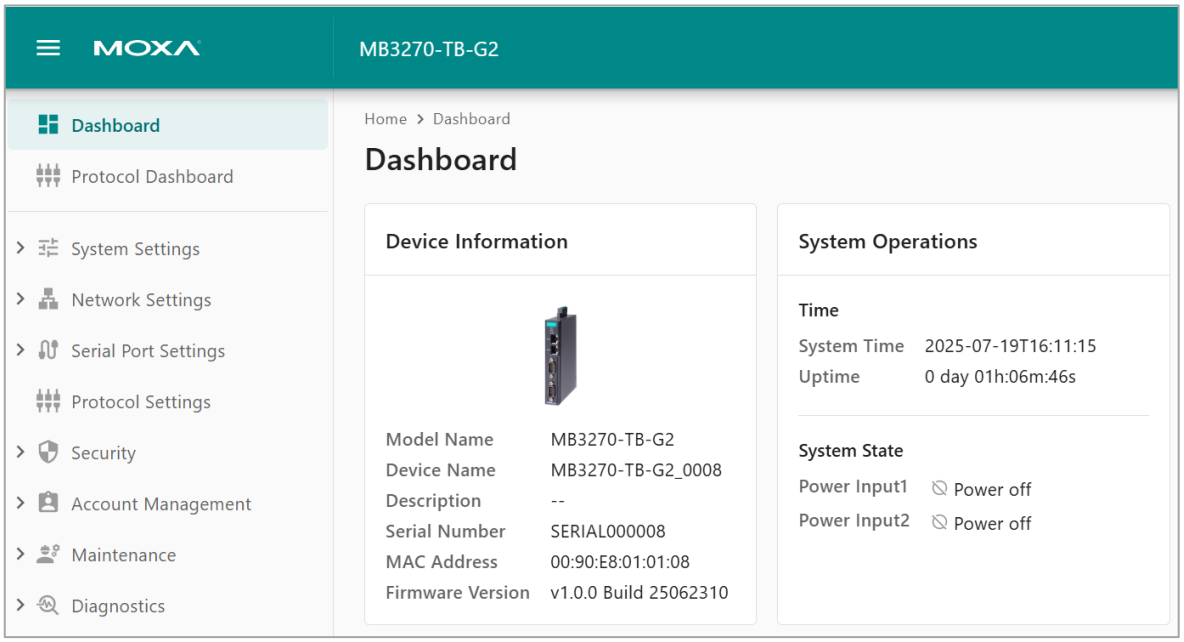
How many categories you may see on the Navigation Panel depends on the privilege of the user group you belong to. The administrators will see all of them as in snapshot above.

Web Console Navigation

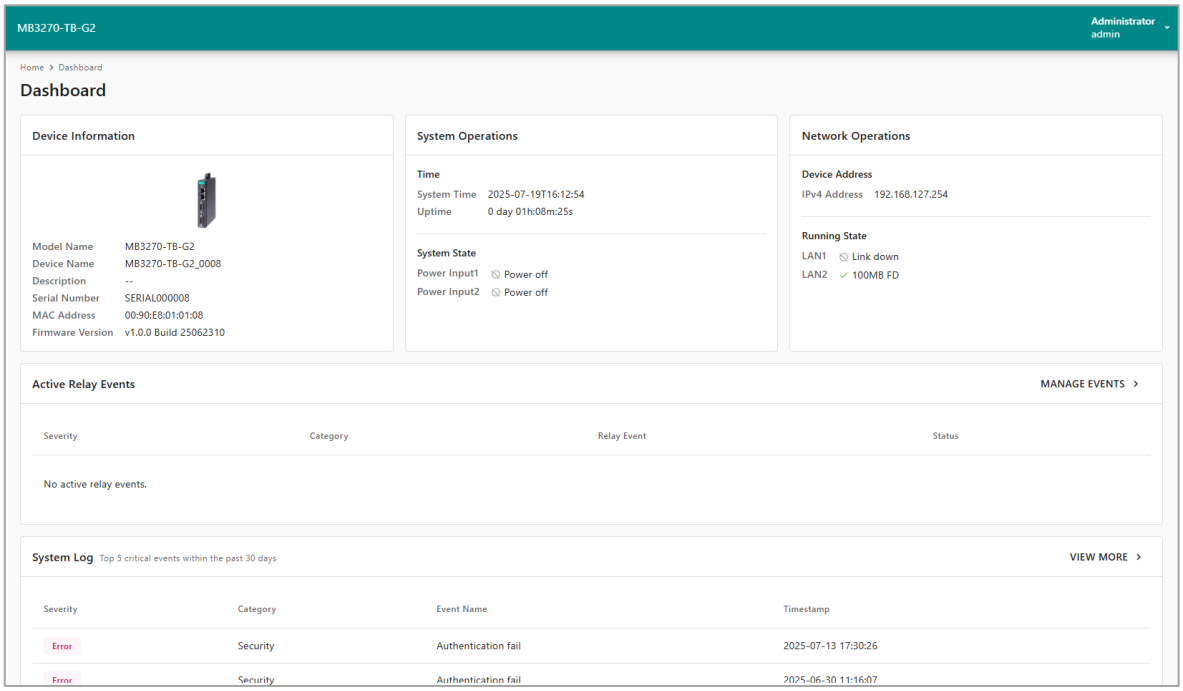
The MGate MB3000-G2 web console features a left-side navigation panel with an expandable menu tree for accessing various settings. Selecting a menu item displays its corresponding options in the main window, where configuration changes can be made. Most changes take effect immediately without a reboot; however, IP address changes require a reboot to take effect because it will require notifying all network devices and updating their network related tables.

Dashboard

The MGate MB3000-G2 features two dashboards: the System Dashboard and the Protocol Dashboard. The System Dashboard provides an overview of key device statuses, while the Protocol Dashboard displays important communication statuses, including Modbus TCP/RTU/ASCII, Real COM, and proprietary serial protocols.



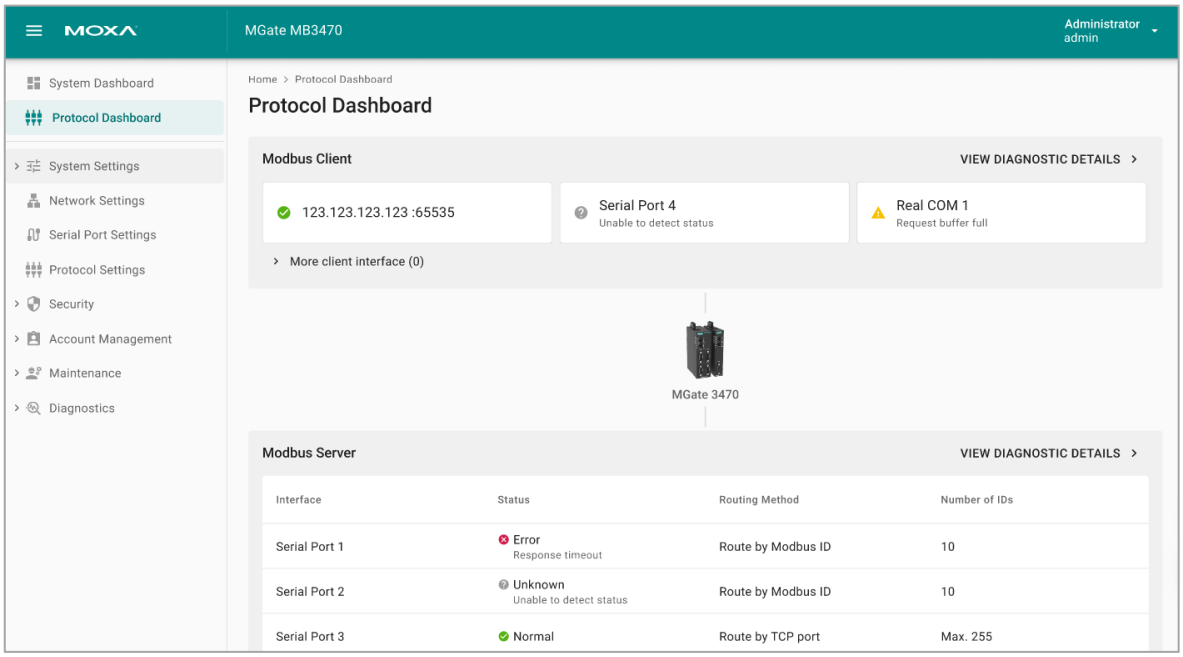
System Dashboard



The System Dashboard summarizes the MGate's key device information, organized into five sections:

1. **Device Information:** Displays basic device information, including the Model Name, Serial Number, MAC address, and firmware version.
2. **System Operations:** Shows information about the unit's operation, such as the device's power-on time, system time, and power input status.
3. **Network Operations:** Displays the unit's network status, including IP address(es), Ethernet LAN status and speed, and any abnormal network events.
4. **Active Relay Events:** Shows active relay events—those currently occurring and requiring attention.
5. **System Log:** Displays any critical events that have occurred since the last login, highlighting any abnormal events.

Protocol Dashboard



The Protocol Dashboard displays important communication statuses for northbound devices (e.g., SCADA or monitoring systems) and southbound devices (such as meters). It shows the following information:

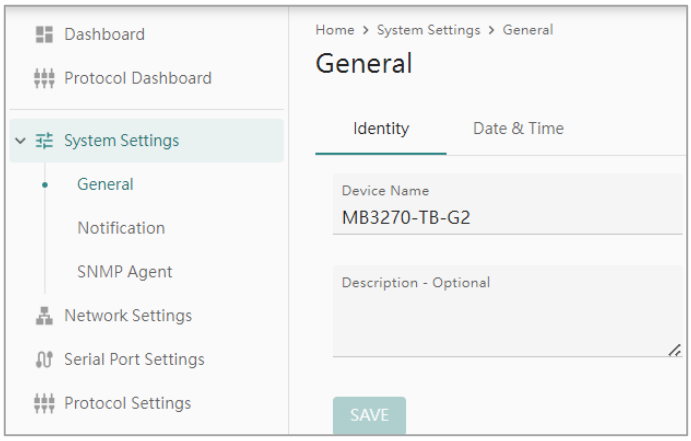
1. A basic system topology of the MGate.
2. Communication protocols and roles (e.g., Modbus TCP/RTU/ASCII, Real COM, and proprietary serial protocols), including client and server roles.
3. Communication statuses: Unknown, Normal, Warning, or Error.

For more detailed diagnostic information, a quick link directs you to the related diagnostics page for further troubleshooting.

System Settings

The first category of the navigation panel is System Settings, which includes three parts. The General page has the Identity and Date & Time settings of the device. The Notification page has the system events, emails, and SNMP Trap/Inform settings. The SNMP Agent has the SNMP Agent settings, which will be needed if you want to get information or settings from the MGate MB3000-G2 device via SNMP protocol.

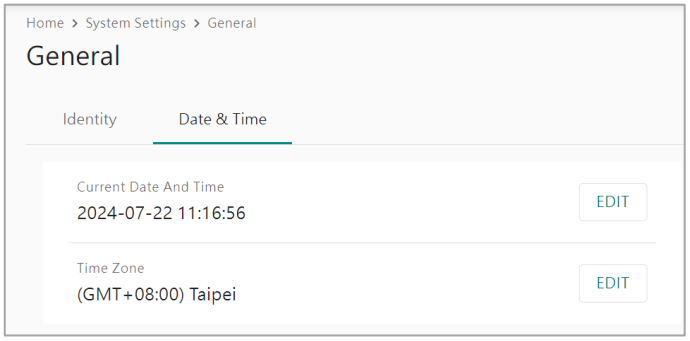
General



Under the General page, the Identity tab provides the Device Name and Description column for you to identify which unit the MGate MB3000-G2 is using.

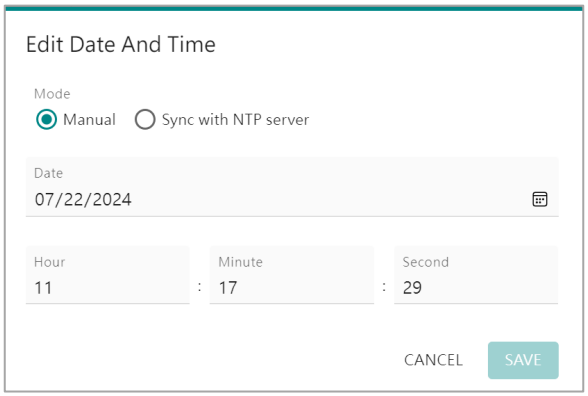
Device Name: This is an optional free text field for your own use. It does not affect the operation of the MGate MB3000-G2. It will be set as the Model Name of the device and the last 4 digits of the serial number. It helps differentiate one MGate MB3000-G2 server from another.

Description: This is an optional free text field for your own use. It does not affect the operation of the MGate MB3000-G2. It is useful for assigning or describing the location of an MGate MB3000-G2. In a network environment of multiple servers, this can be a valuable aid when performing maintenance.



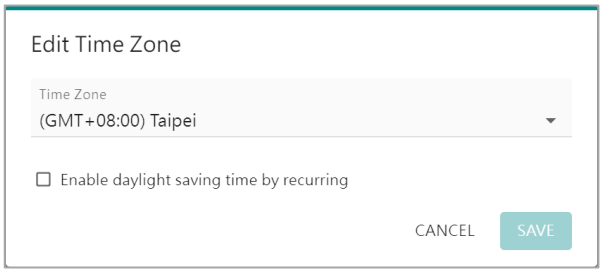
The MGate MB3000-G2 has a built-in Real-Time Clock for time calibration functions. To change the time, please switch to the Date & Time tab. Select the **EDIT** button to change the current date and time and the time zone.

The MGate MB3000-G2 uses SNTP (RFC-1769) for auto time calibration. Enter a time server IP address or domain name in this optional field. Once the correct time server address is set, the MGate MB3000-G2 will regularly request time information from the time server every 10 minutes.

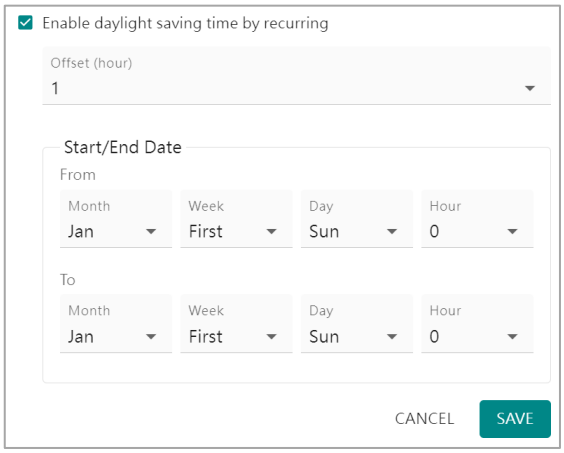


To change the time zone, select the **EDIT** button and select the location of the device.

The time zone will be adjusted automatically.



If daylight saving time applies in the summer, enable the checkbox **Enable daylight saving time by recurring**.



Daylight saving time (also known as **DST** or **summertime** involves advancing clocks (usually one hour) during the summer to provide an extra hour of daylight in the afternoon.

Offset

Setting	Description	Factory Default
User adjustable hour	The clock should be set forward by the number of hours specified in the offset parameter.	1

Start Date

Setting	Description	Factory Default
User adjustable date	The Start Date parameter allows users to enter the date that daylight saving time begins.	The Sunday of the First week of January

End Date

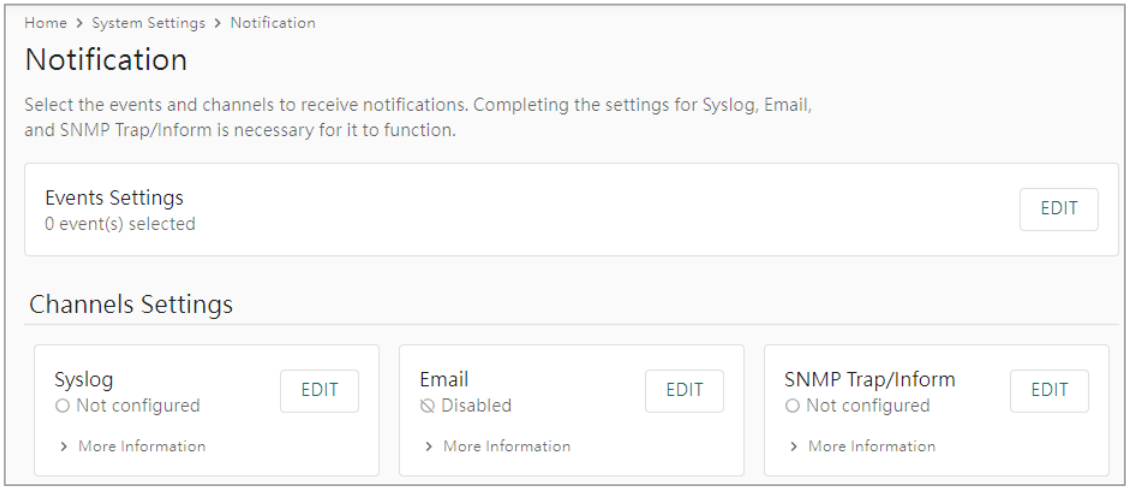
Setting	Description	Factory Default
User adjustable date	The End Date parameter allows users to enter the date that daylight saving time ends.	The Sunday of the First week of January

ATTENTION

A risk of an explosion exists if the real-time clock battery is replaced with the wrong type!

The MGate MB3000-G2's real-time clock is powered by a lithium battery. We strongly recommend that you do not attempt replacement of the lithium battery without help from a qualified Moxa support engineer. If you need to change the battery, please contact the Moxa RMA service team.

Notification



Notification Settings allow you to customize events that are logged by the MGate MB3000-G2. Events are grouped into five categories, known as event groups. Select which groups or events you want to log in to the **Remote Log** server. An email or SNMP Trap/Inform can also notify the administrator immediately of some of the events.

By default, the MGate will enable the event severity as Notice, Warning, and Error under the Security category and save them on the local flash memory. For the local log settings, find the diagnostics section. If you have a central management log server, configure the relative settings under **System Settings > Notification**.

Local Log	Keep the log in the flash of MGate MB3000-G2 up to 10,000 items.
Remote Log	Keep the log in the remote defined Log Server. You will need to assign a remote Log Server in the System Management/Misc. Network Settings/Remote Log Settings if a remote log is checked.

The Categories of Notifications

Setting	Description
System	The events related to the MGate itself, like firmware ready.
Network	The events related to the Ethernet interface, for example, the Ethernet link up.
Security	The events which may be considered security related; the administrator may need to figure out why it happened. For example, a Login fail event.
Maintenance	The events which usually happen at maintenance process, for example, firmware upgrade.
Serial	The events related to the serial interface(s), for example, Port connect.

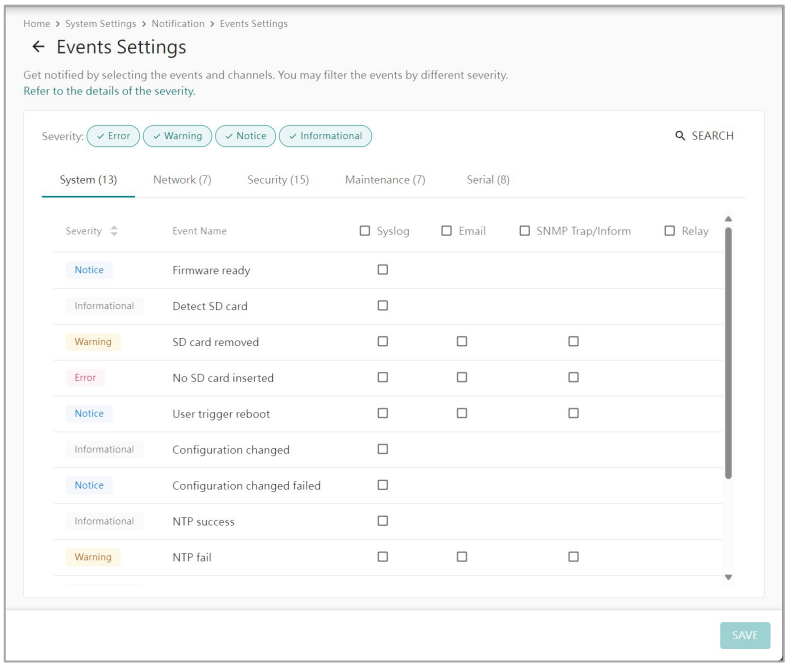
The Severity of Events

Based on RFC5424, the severity of different events is categorized according to the following priority and description.

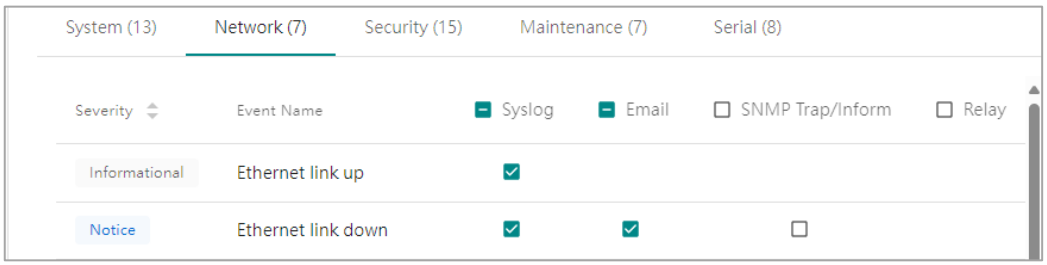
Priority	Severity	Description
1	Error	Events that indicate problems, but in a category that does not require immediate attention.
2	Warning	Events that provide forewarning of potential problems indicate that some further action could result in a critical error.
3	Notice	Events that are not error conditions, but that may require special handling.
4	Informational	Confirmation that the program works as expected.

The logs are essential for troubleshooting in case of errors. Refer to Appendix C for a detailed event list.

Event Settings

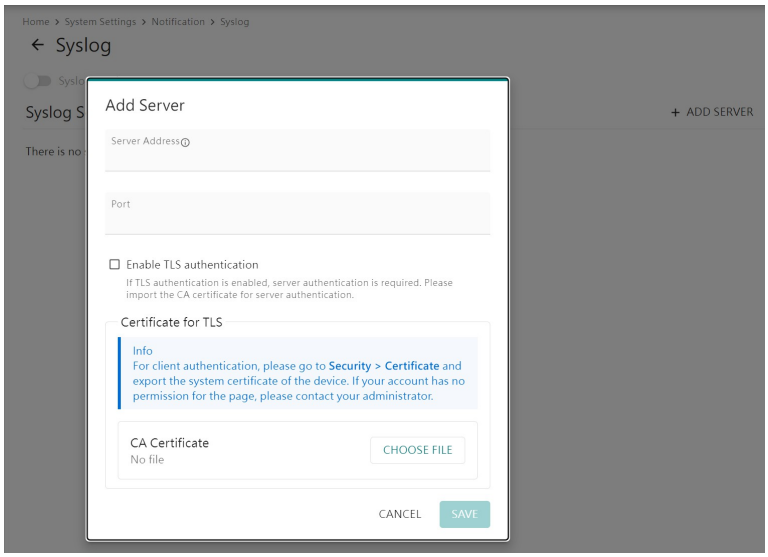


When selecting the **EDIT** button of the **Events Settings** column, you will see the event list, separated into different categories. Select the checkbox to enable the event for Syslog, Email, SNMP Trap/Inform, or the Relay function. Only the enabled events will be recorded on the Syslog or trigger an email, SNMP Trap/Inform, or Relay output.



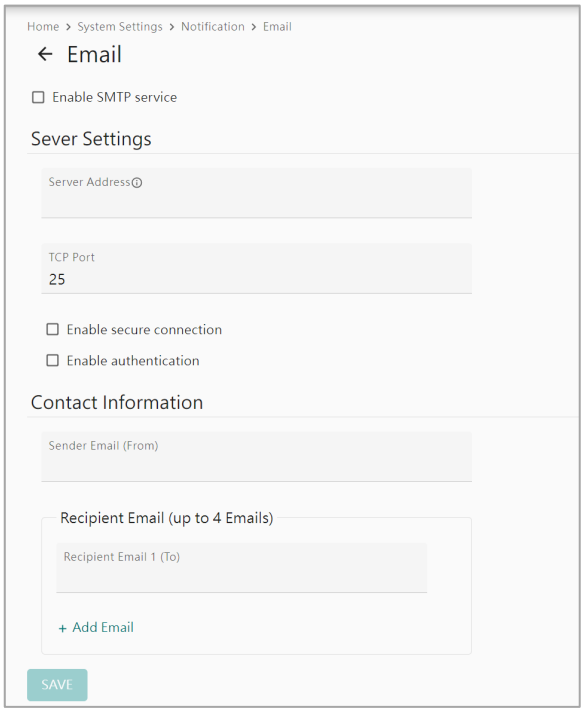
Channels Settings

Once you choose which events to record, set up the recording location and decide on email, SNMP Trap/Inform, or Relay for immediate notification.



Select the **EDIT** button in the Syslog column. Input the remote log server's IP address and port to receive the events from the MGate. You can also enable TLS authentication and import a CA certificate to secure communication for the log recording.

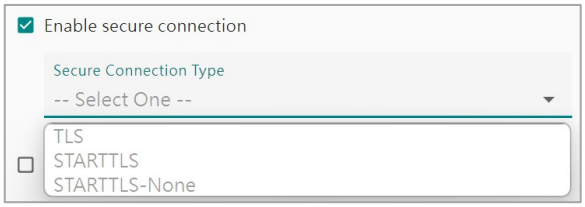
Email



Select the **EDIT** button in the Email column. Enable the SMTP service so that the MGate will send an email if the selected events happen.

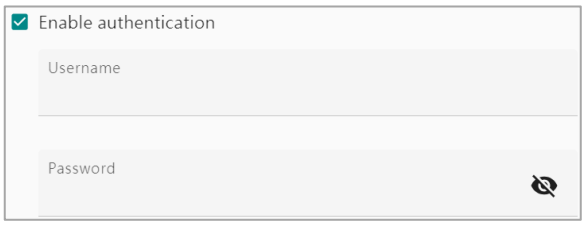
Server Settings

Setting	Description	Factory Default
Server Address	The IP address or domain name of the SMTP server.	N/A
TCP port	The TCP port to which the SMTP server receives SMTP messages.	25



If the SMTP server requires a secure connection (encrypt the email), select **Enable secure connection**. There are three options.

Setting	Description	Factory Default
TLS	Encrypts the entire communication channel between the client and the server from the beginning, ensuring that all data transmitted is secure.	N/A
STARTTLS	It is possible to start the connection in plain text and then switch to encrypted mode through STARTTLS. If the upgrade fails, the communication remains in plain text.	N/A
STARTTLS-None	No encryption. STARTTLS-None as an option helps system administrators clearly specify which connections should remain unencrypted.	N/A



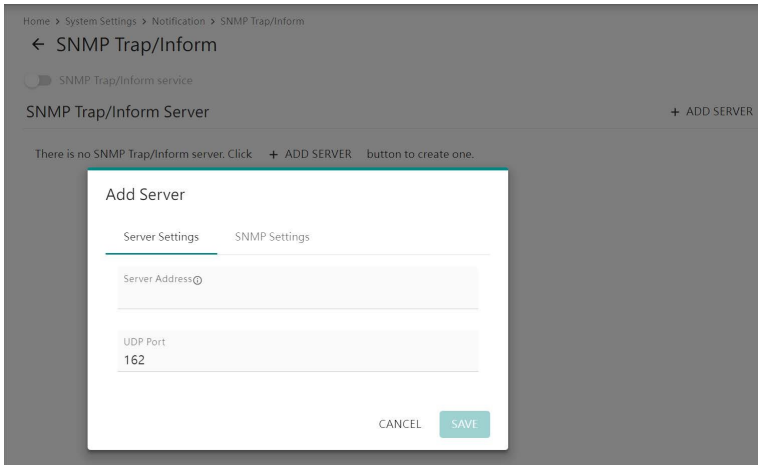
If the SMTP server requires authentication verification, select **Enable authentication**, and input the username and password used to log into the SMTP server.

Setting	Description	Factory Default
Username	The name used to log into the SMTP server.	N/A
Password	The password is used to log into the SMTP server.	N/A

Contact Information

Setting	Description	Factory Default
Sender Email (From)	The email address that the MGate will use to send the message. The user can easily figure out which MGate sends the message by this account.	N/A
Recipient Email 1 (To)	The email address that the MGate will send the message to. It shall be the administrator/manager of the MGate who manages/monitors the status of the MGate or the serial device connected to the MGate. There are at most four recipient emails .	N/A

SNMP Trap/Inform

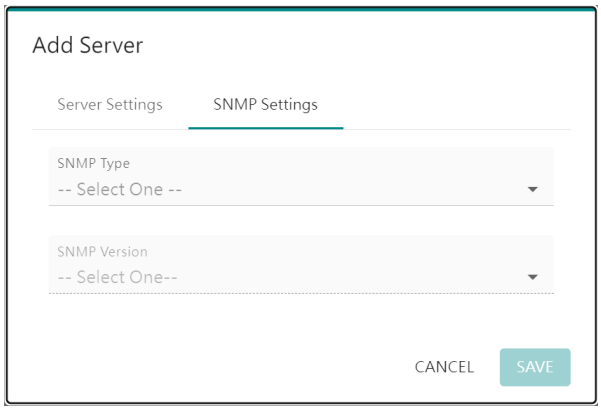


Select the **EDIT** button at SNMP Trap/Inform column and select **ADD SERVER**. Set the Server Setting and the SNMP Settings.

Server Settings

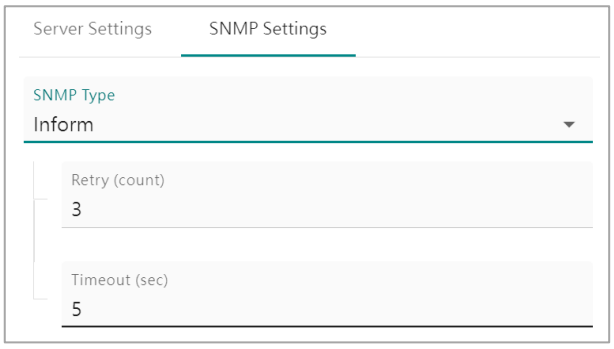
Setting	Description	Factory Default
Server Address	The IP address or domain name of the SNMP server.	N/A
UDP port	The UDP port at which the SNMP server receives SMTP messages.	162

SNMP Settings



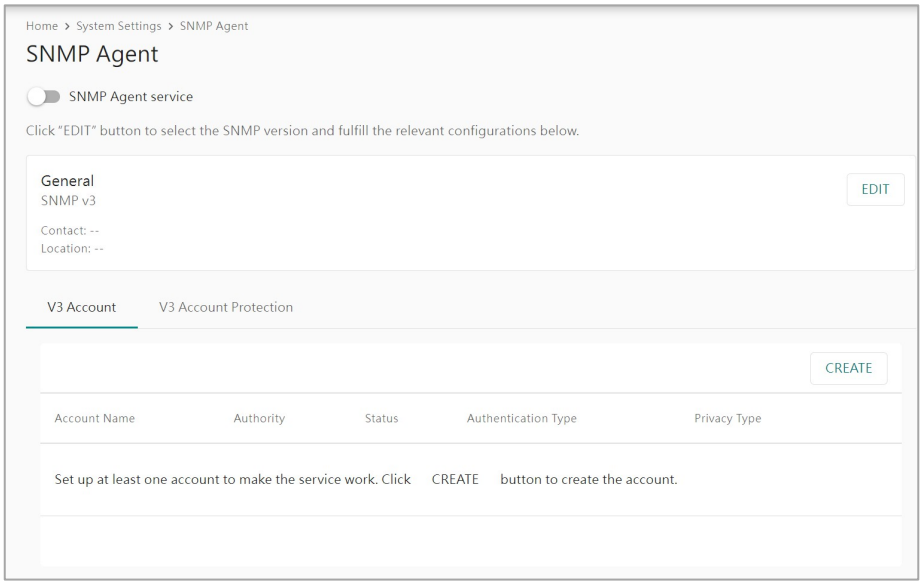
SNMP Type	Description	Retry (Count)	Timeout (sec)	SNMP version
Trap	The MGate will send SNMP Trap and will not wait for acknowledgment	N/A	N/A	v1/v2c/v3
Inform	After sending an SNMP Inform, the MGate waits for the acknowledgment. The MGate will resend the Inform message until it gets a confirmation or times out.	Number of retries Default=3	The duration before a timeout occurs. Default=5	v2c/v3

SNMP Inform messages require acknowledgement of notifications. If you choose SNMP Inform as the SNMP type, you might have to specify the number of retries the MGate should attempt if it doesn't receive acknowledgments. Also, determine the time interval for the MGate to wait before sending the SNMP Inform message.


Server Settings
SNMP Settings
SNMP Type
Inform
Retry (count)
3
Timeout (sec)
5

SNMP Agent

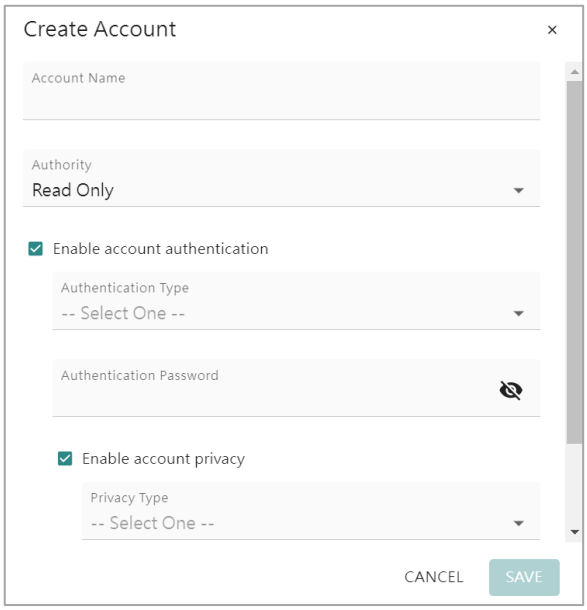
Simple Network Management Protocol (SNMP) is a widely used protocol/tool for network administrators to manage and monitor network devices. To meet this requirement, the MGate MB3000-G2 Series supports SNMPv1/v2c/v3 and includes a private MIB for device management and status monitoring of Ethernet or serial communication. For such purposes, enable the SNMP Agent service here (**System Settings > SNMP Agent**) and configure the proper settings introduced in the following sections.



Select the **EDIT** button under the General column. Select the SNMP Version and set the Device Details.

Setting	Description	Factory Default
SNMP Version	Select the SNMP Version. Use only SNMP v3/Use only v1, v2c/Use v1, v2c, and v3.	v3
Contact - Optional	This field usually includes an emergency contact name and telephone or pager number.	N/A
Location - Optional	Use this field to specify the location string for SNMP agents such as the MGate MB3000-G2. This string is usually set to the street address where the MGate MB3000-G2 is physically located.	N/A

When using SNMP v3, you need to create a V3 Account first. Select the **CREATE** button in the V3 Account column.



Account Name: Use this field to identify the username for the specified level of access.

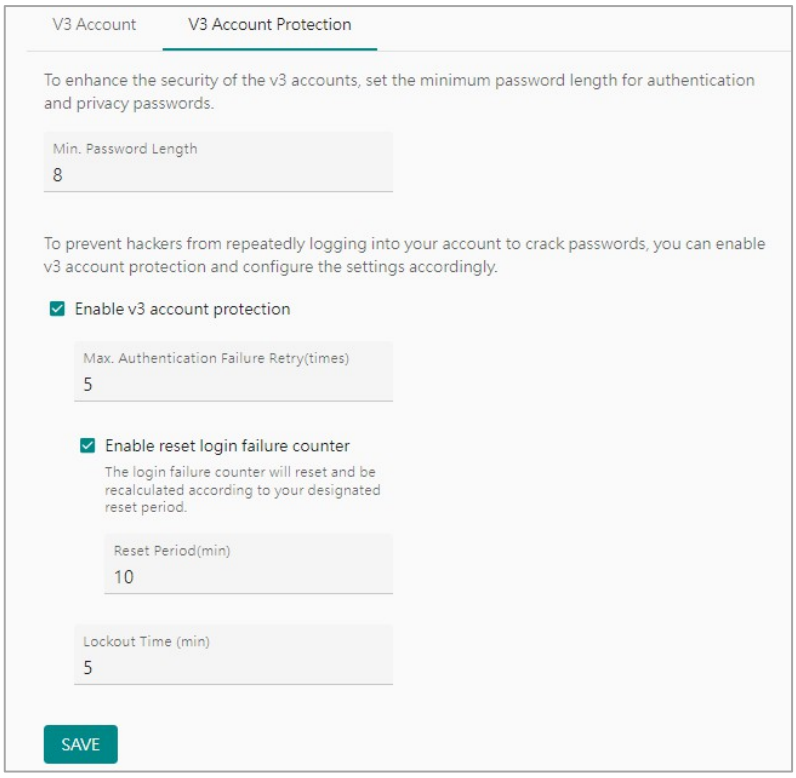
Authority: Select authentication parameters for two levels of access: Read Only(default) and Read/Write.

When enabling account authentication, select the Authentication Type and input the Authentication Password.

Authentication Type: Use this field to select MD5 or SHA as the method of password encryption.

Authentication Password: Use this field to set the password.

Privacy Type: Use this field to enable DES_CBC or AES_128 data encryption when you enable account privacy.



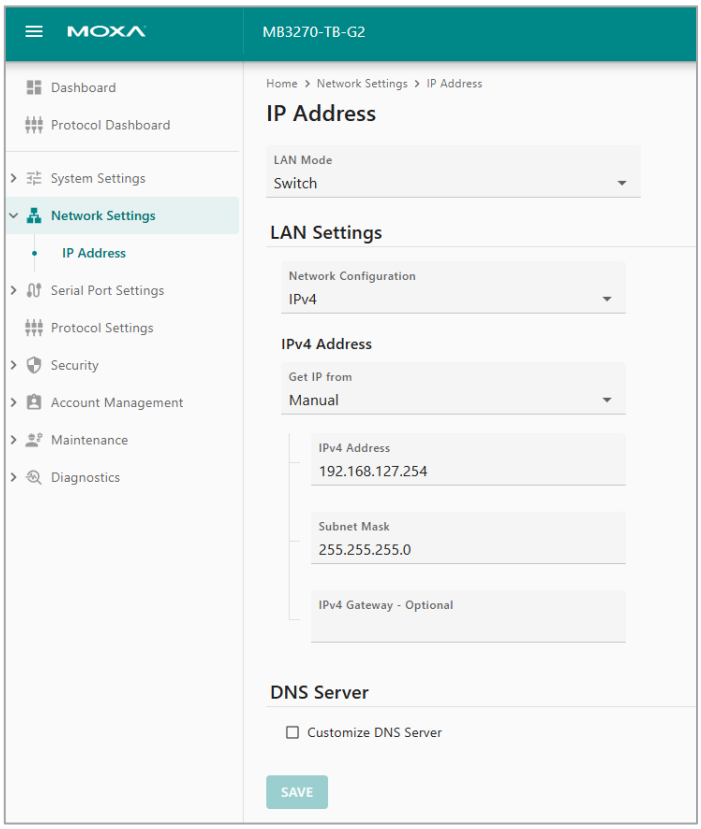
Go to the **V3 Account Protection** tab to set the minimum password length for authentication and privacy passwords. Enable v3 account protection can set the maximum authentication failure times and lockout time. Additionally, you can enable the reset login failure counter to automatically reset and recalculate it within your designated reset period.

Network Settings

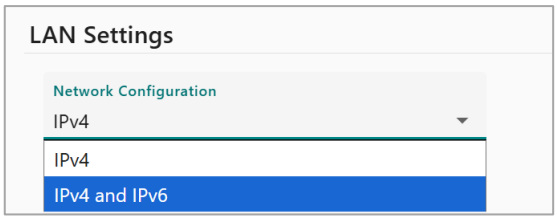
IP Adress

Network devices require IP addresses to communicate. This address should have been configured during the initial login. In the Network Settings category, you can adjust advanced settings or change the existing IP address.

MGate MB3x70-G2 provides flexibility to configure different subnets on the two LANs. Refer to the Dual IP parameter below.



Network Configuration

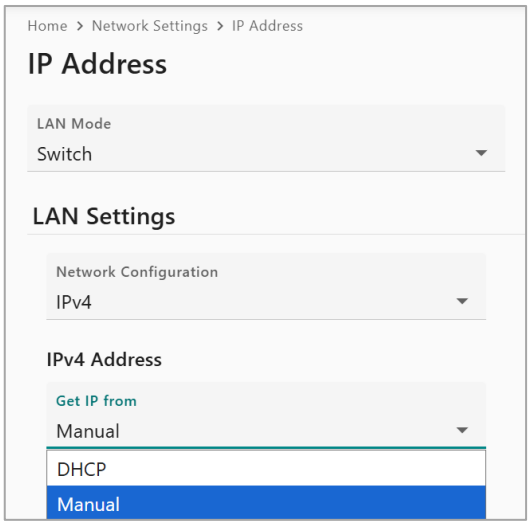


What Is IPv6 Address?

The abbreviation IPv6 stands for Internet Protocol version 6. IPv6 is the second version of the Internet Protocol, introduced after IPv4. What distinguishes the two versions is the varying lengths of the IP addresses. IPv4 uses 32-bit IP addresses; IPv6 uses 128-bit IP addresses. IPv4 is still the most widely used protocol on the Internet. If your devices and network infrastructure are limited to IPv4 compatibility, opt for IPv4 only. However, if you have already deployed IPv6 and need many IP addresses, then select IPv4 and IPv6.

IPv4 Address

Get IP From: **DHCP** or **Manual**. If there is a DHCP server on the network that assigns the IP address automatically, select **DHCP**. If not, select Manual and input the IPv4 address, subnet mask, and IPv4 gateway.

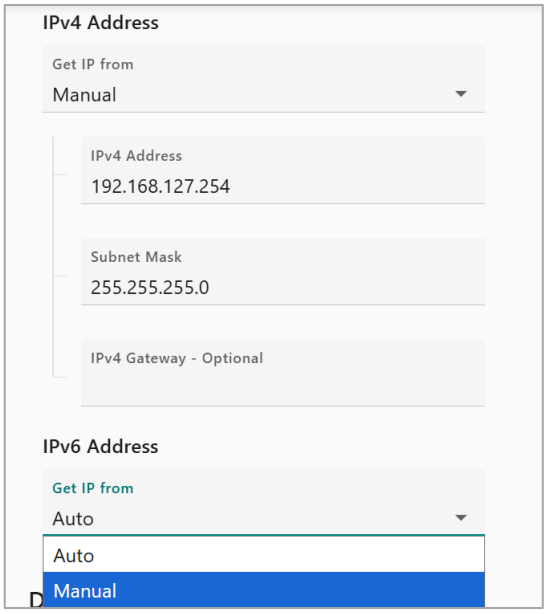


IPv4 Address (default=192.168.127.254): Enter the IP address that will be assigned to your MGate MB3000-G2. All ports on the MGate MB3000-G2 will share this IP address. An IP address is a number assigned to a network device (such as a computer) as a permanent address on the network. Computers use the IP address to identify and talk to each other over the network. Choose a proper IP address that is unique and valid in your network environment.

Subnet Mask (default=255.255.255.0): Enter the subnet mask. A subnet mask represents all the network hosts at one geographic location, in one building, or on the same local area network. When a packet is sent out over the network, the MGate MB3000-G2 will use the subnet mask to check if the desired TCP/IP host specified in the packet is on the local network segment. If the address is on the same network segment as the MGate MB3000-G2, it establishes a connection directly. Otherwise, the connection is established through the default gateway.

IPv4 Gateway: Enter the IP address of the gateway if applicable. A gateway is a network computer or device that acts as an entrance to another network. Usually, the devices that control traffic within the network or at the local Internet service provider are gateway nodes. The MGate MB3000-G2 needs to know the IP address of the default gateway device to communicate with the hosts outside the local network environment. For correct gateway IP address information, consult the network administrator.

IPv6 Address

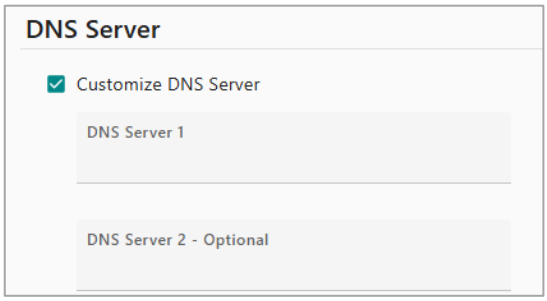


Get IP From: You can choose from two possible IP configuration modes, Auto or Manual.

IPv6 Address: Enter the IPv6 address that will be assigned to your MGate MB3000-G2. All ports on the MGate MB3000-G2 will share this IPv6 address. An IPv6 address is a number assigned to a network device (such as a computer) as a permanent address on the network. Computers use the IPv6 address to identify and talk to each other over the network. Choose a proper IPv6 address that is unique and valid in your network environment.

Prefix: The prefix is the part of the address that shows the bits that have fixed values or are the bits of the subnet prefix. Prefixes for IPv6 subnets, routes, and address ranges are expressed in the same way as Classless Inter-Domain Routing (CIDR) notation for IPv4. An IPv6 prefix is written in address/prefix-length notation. For example, 21DA:D3::/48 and 21DA:D3:0:2F3B::/64 are IPv6 address prefixes.

DNS Server



Domain Name System (DNS) is responsible for translating internet domain names into IP addresses. A domain name is an alphanumeric name, such as www.moxa.com, which is easier to remember than the numerical IP address. A DNS server is a host that translates this kind of text-based domain name into the actual IP address used to establish a TCP/IP connection. When a user wishes to access a specific website, their computer sends the domain name (e.g., moxa.com) to a DNS server to obtain the website's IP address. The user's computer connects to the website's web server using the IP address obtained from the DNS server.

The MGate MB3000-G2 acts as a DNS client, actively querying the DNS server for domain name IP addresses. The following functions on the MGate MB3000-G2 web console support the use of domain names in place of IP addresses: Time Server, Destination IP Address (in TCP Client mode), Mail Server, SNMP Trap Server, Destination Address (in Pair Connection mode), Primary/Secondary Host Address (in Terminal mode), RADIUS Server, TACACS+ Server and SMTP Server.

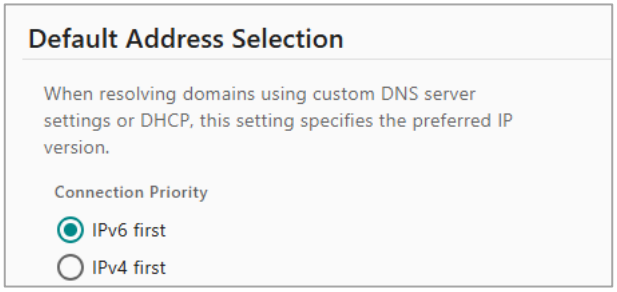
DNS server 1: Choose Customize DNS server to enter the DNS server's IP address in this field. This allows the MGate MB3000-G2 to use domain names instead of IP addresses to access hosts.

DNS server 2: This is an optional field. The IP address of another DNS server can be entered in this field if DNS server 1 is unavailable.

Under **IP Address**, you can configure the MGate's LAN mode, IP address, netmask, and gateway.

Parameter	Value	Description
LAN Mode	Switch (default), Dual IP	Switch mode allows users to install the device with daisy chain topology. Dual IP mode allows the gateway to have two different IP addresses, each with distinct netmask and gateway settings. The two IP addresses share the same MAC address.
IPv4 Address—Get IP From	Manual, DHCP	Select "Manual" if you are using a fixed IP address. Select the DHCP option if you want the IP address to be dynamically assigned.
IPv4 Address	192.168.127.254 (or other 32-bit number)	The IP Address identifies the server on the TCP/IP network.
Subnet Netmask	255.255.255.0 (or other 32-bit number)	A number that helps network devices understand how an IP address is divided. It identifies which network MGate belongs to.
IPv4 Gateway	Optional field, or other 32-bit number such as 192.168.127.255	The IP address of the router that provides network access outside the server's LAN.
DNS Server 1	Optional field, or other 32-bit number such as 192.168.127.1	The IP address of the primary domain name server.

Default Address Selection

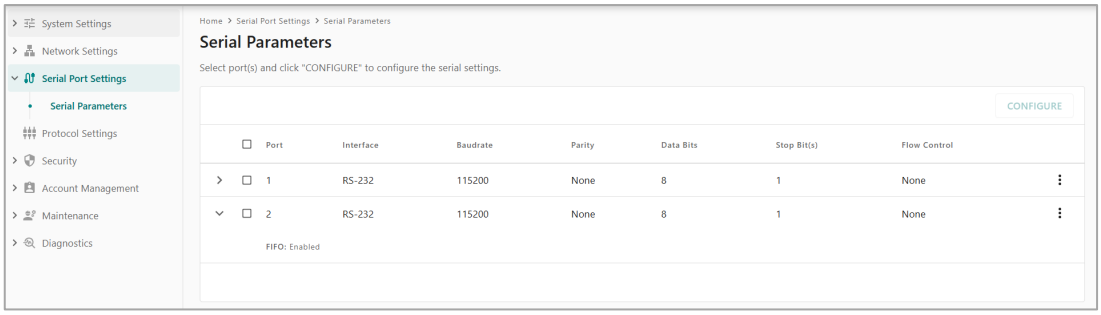


When selecting 'IPn4 and IPn6' under Network Configuration, you can define the connection priority. This feature works with the MGate MB3000-G2 functions that use the domain name to get the IP address of the remote host/server. For this kind of application, the MGate MB3000-G2 will ask for the IP address of the remote host/server through the DNS. The DNS will reply with both the IPv4 and IPv6 IP addresses if both exist simultaneously in the remote host/server. For this reason, you need to define which one has a higher priority, IPv6 first (RFC 3484) or IPv4 first.

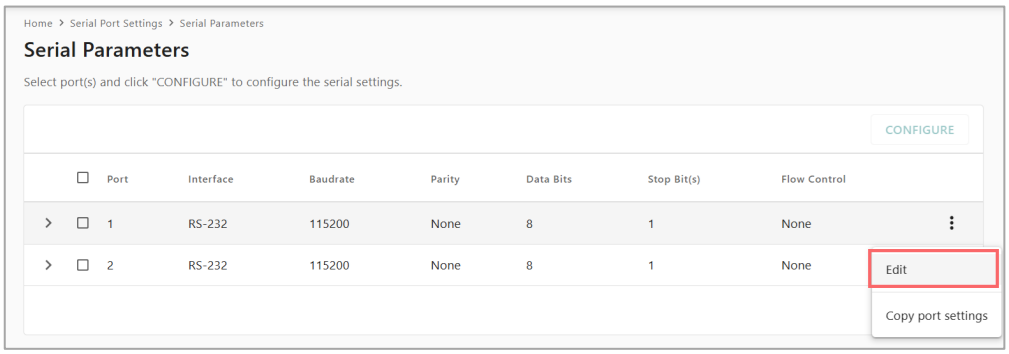
Serial Port Settings

Serial Parameters

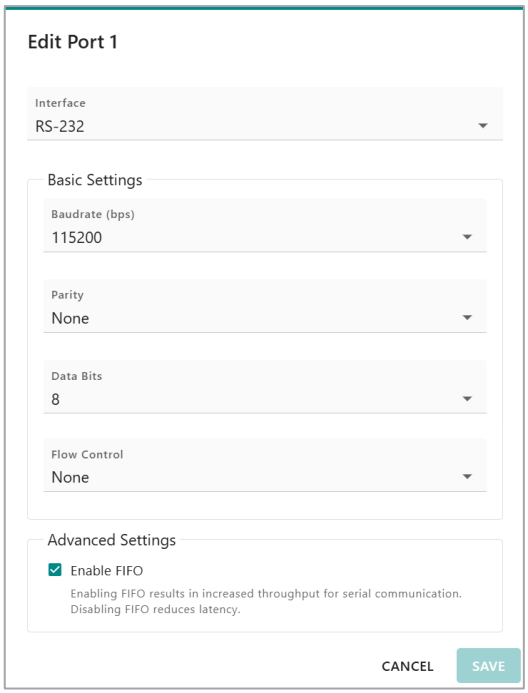
For Modbus serial communication to function correctly, the serial parameters of the serial device and the MGate's serial port must match. Refer to the manual for the connected Modbus or serial device to determine its serial parameters. Then, configure the MGate's serial port by navigating to **Serial Port Settings > Serial Parameters** within the MGate's interface.



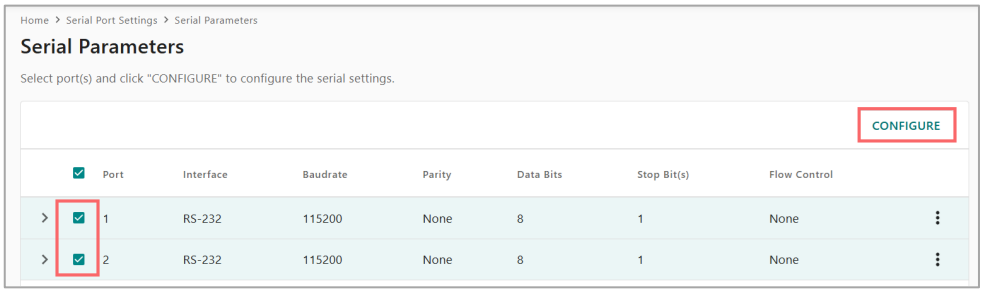
Select the ⋮ button and EDIT to change the serial parameters for a specific serial port. The Edit Port window will open.



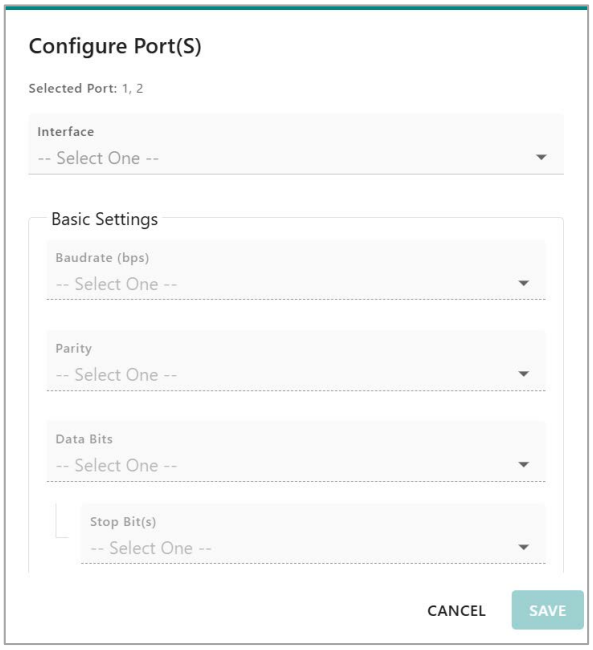
Change the serial port settings in the Edit Port window.



To configure multiple serial ports at once, select the checkboxes of the target ports and select the **CONFIGURE** button.



The Configure Port(s) window will then open, allowing you to set new values for all selected ports.

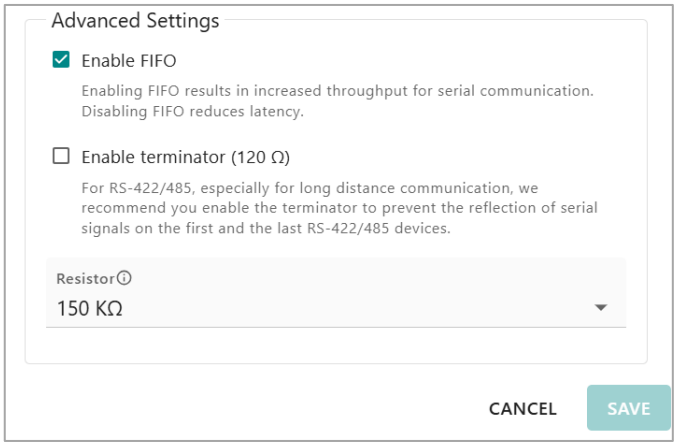


Basic Settings

For basic serial port parameters definition, refer to the table below.

Parameter	Value	Description
Interface	**RS-232(default), RS-422, RS-485 2-wire, RS-485 4-wire**	You may configure the serial interface to RS-232, RS-422, RS-485 2-wire, or RS-485 4-wire, depending on your connected serial device(s)
Baudrate	Default=115200, Selectable standard baudrates (bps) under Modbus RTU/ASCII protocol: 50/75/110/134/150/300/600/1200/1800/2400/4800/7200/9600/19200/38400/57600/115200/230.4k/460.8k/921.6k Self-defined baudrates are supported under Proprietary Serial protocol.	For Modbus serial protocols, this field allows you to select standard baud rates from the drop-down menu. For proprietary serial protocols, you can configure custom baud rates in this field.
Parity	None(default), Odd, Even, Mark, Space	This field configures the parity parameter.
Data Bits	8	This field configures the data bits parameter. For Modbus protocol, this field should only be 8.
Stop Bit(s)	1(default), 2	This field configures the stop bits parameter; 1 or 2 are supported.
Flow Control	None(default), RTS/CTS, RTS Toggle	This field configures the flow control type, including RTS/CTS, DTR/DSR, Xon/Xoff, RTS Toggle and None. When setting the interface as RS-232, it supports all the above flow control mechanisms.

Advanced Settings

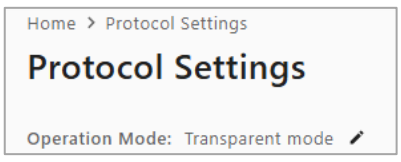


For advanced serial port settings, refer to the table below.

Parameter	Value	Description
Enable FIFO	Enable (Default), Disable	The Enable FIFO function is enabled by default to improve data throughput. There are two situations where the user may choose to disable this function by unchecking the checkbox. • If the serial device does not have FIFO/buffer or does not support flow control function. In this case, the serial device may not receive the serial data from the MGate on time, which means that some data might be dropped. • If the data latency is more important than data throughput. To achieve higher data throughput, data can be temporarily stored in the buffer, allowing for larger amounts of data to be sent at once. The downside is that the latency of a single data may be slower. If the latency is important for the serial device to read data correctly, then you should consider disabling the Enable FIFO function.
Enable terminator (120 Ω)	Disable (default), Enable	When configuring the interface as RS-422, RS-485 2-wire, or RS-485 4-wire, you can choose to enable the terminator (120 Ω) and set the resistor. Because these interfaces can handle communication distances of over 1 km and accommodate over 10 serial devices on the same bus, there are more factors that need to be considered. For long-distance communication, we recommend you enable the terminator to prevent the reflection of serial signals on the first and the last RS-422/485 devices.
Resistor	150 KΩ (default), 1 KΩ	If the remote devices are unable to receive data correctly for RS-422/485, try adjusting the pull high/low resistors which can strengthen the serial signal, and it might help with this. Two values are selectable, 1 KΩ or 150 KΩ.

Protocol Settings

The MGate MB3x70-G2 provides two operation modes for Modbus communication: **Transparent mode** and **Agent Mode**. In Transparent mode, the gateway will bypass and translate Modbus commands between Modbus TCP and RTU/ASCII. In Agent mode, the gateway will actively poll the Modbus server devices and store the data in the gateway's memory. The Modbus client can retrieve Modbus server devices' data via the gateway's memory.



When to use Transparent Mode:

When you want the Modbus gateway to act as a **pure protocol converter** with no device ID/address mapping. Each Modbus TCP client communicates **directly with the Modbus RTU/ASCII server** (1 request gets 1 response).

Ideal for simple setups where the client already knows the Modbus IDs and addresses and doesn't need to change.

Pros:

- **Simple configuration:** Minimal setup — just map which Modbus TCP or serial interface should the IDs be forwarded/routed out.
- **Direct control:** Modbus clients see the Modbus servers as if they are directly and transparently connected.
- **Limitations:** One Modbus request is passed directly to one Modbus server; no data aggregation or advanced mapping.

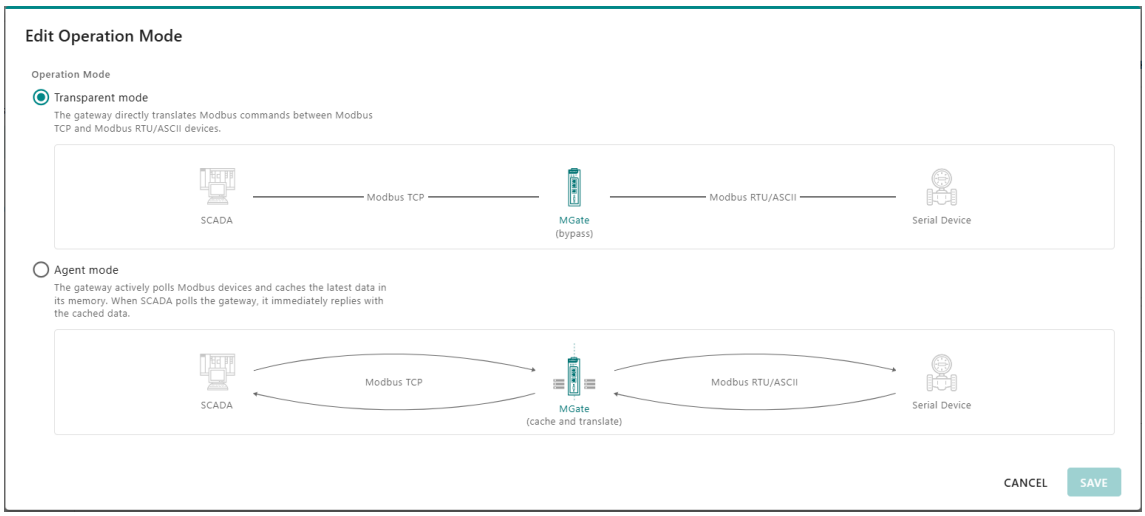
When to use Agent Mode:

When you need the gateway to **act as a Modbus device itself**, or when you need to **aggregate multiple Modbus devices** into a single device view. In Agent Mode, the MGate actively polls devices and caches data in the MGate. When the Modbus client polls MGate, MGate responds right away, no need to wait for Modbus servers to respond.

Ideal for **SCADA or PLCs that need to aggregate multiple devices into one**, or **the system/Modbus client requires higher performance**.

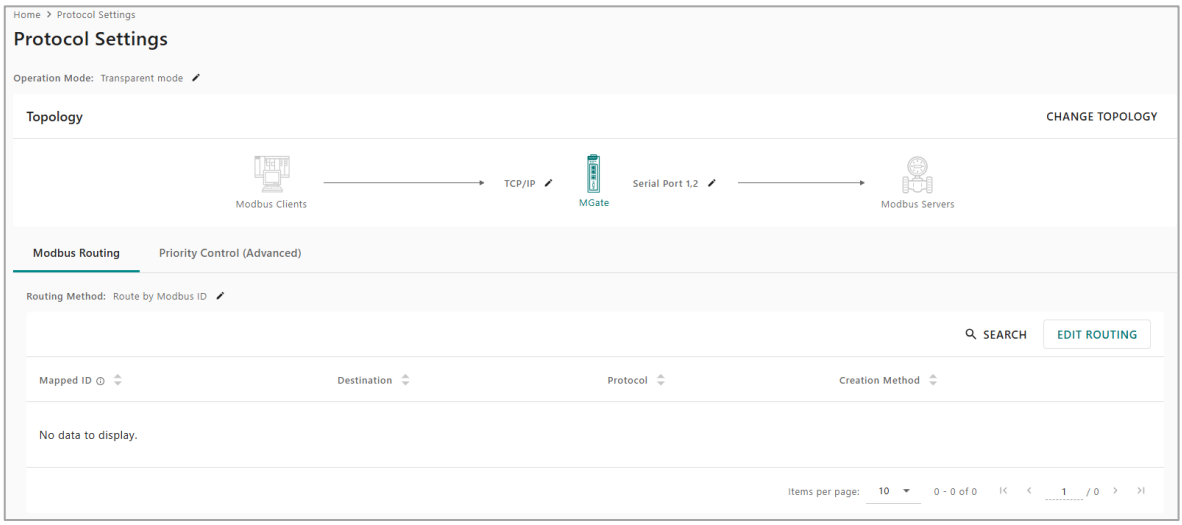
Pros:

- **Aggregation:** Multiple Modbus RTU Servers appear as one consolidated Modbus device.
- **Flexible mapping:** You can reorganize registers (e.g., combine different server addresses into one contiguous address space).
- **Compatibility:** Helps legacy or limited-function Modbus masters communicate with multiple servers.
- **Limitations:** More configuration effort (need to define the commands that will be actively polled from MGate, and the protocol mappings).



Transparent Mode

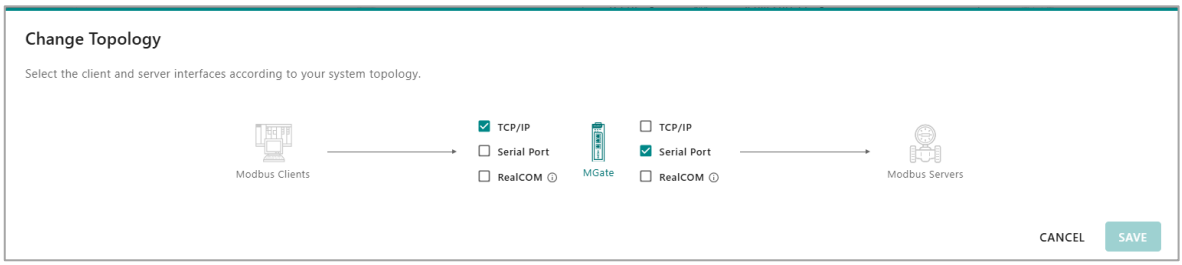
In Transparent Mode, the default topology is the most seen Modbus TCP client—Modbus serial server (RTU). To change the topology, select the **CHANGE TOPOLOGY** button and configure it according to your application topology.



Topology

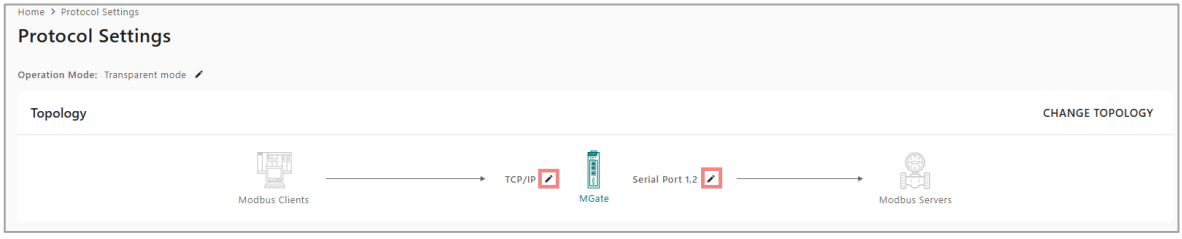
With more than 2 serial ports (MGate MB3270/MB3470-G2 models), Modbus TCP-serial, serial-TCP, TCP-TCP, serial-serial can work at the same time. 1 serial port model (MGate MB3170-G2) is not able to support serial-serial cause data collision cannot be avoided due to physical characteristics.

For applications that run serial-based programs over TCP (virtual COM), Real COM is also supported. Users need to install Windows Driver Manager to map COM ports on their PC. To use Real COM mode, refer to the **Real COM mapping** section.

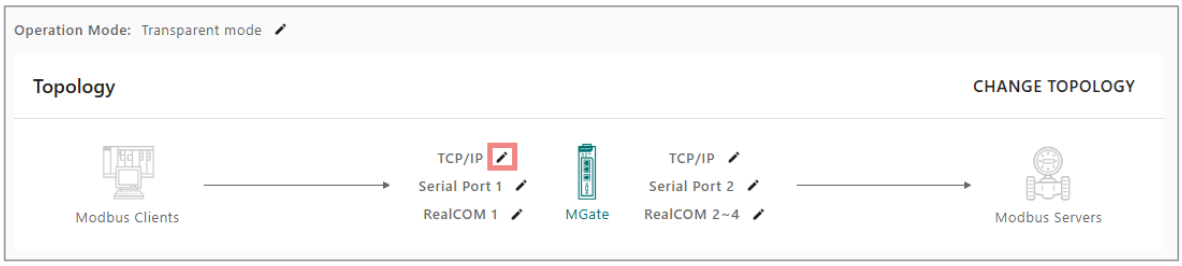


Modbus Settings

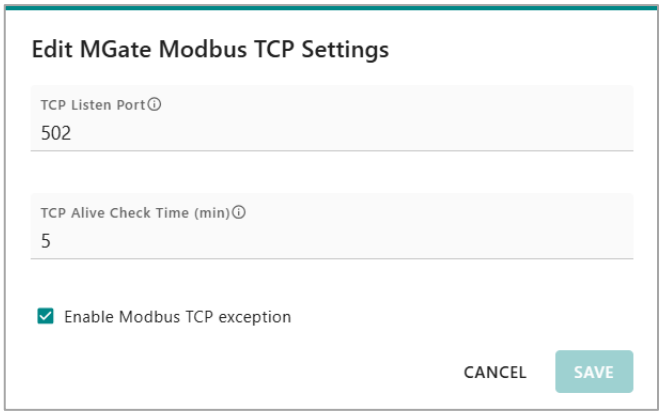
To edit Modbus protocol general settings, select the pen icon besides TCP/IP or Serial Port to change Modbus TCP or Modbus RTU/ASCII settings respectively. According to your topology, there will be different parameters to configure.



Modbus TCP Cient—MGate Modbus TCP Settings

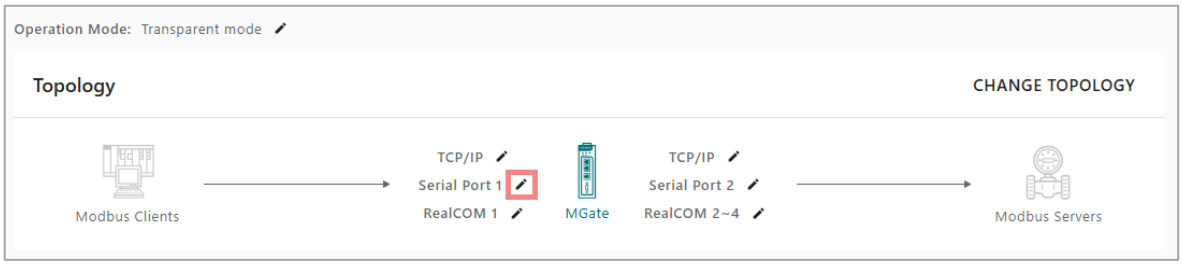


When the MGate listens the Modbus TCP client, the following parameters can be changed according to your system requirements.



Setting	Value
TCP Listen Port	1 to 65535. Allows you to change the Modbus TCP listen port from the default value (502).
TCP Alive Check Time (min)	The MGate will check when it received the last Modbus TCP packet. If it exceeds the timeout (default 5 min), it will reset the TCP session to avoid the TCP session being occupied.
Enable Modbus TCP Exception	Enable or Disable. Default is enabled, so if the MGate detects an error from the Modbus server devices, it will return a Modbus TCP exception to the Modbus client.

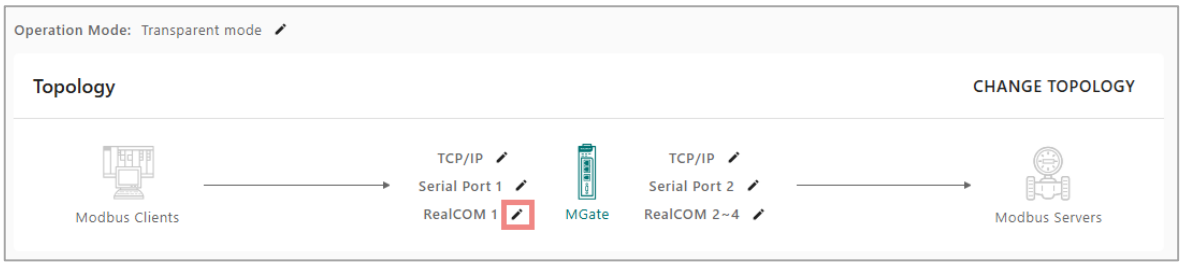
Modbus Serial Client—MGate Modbus Serial Settings



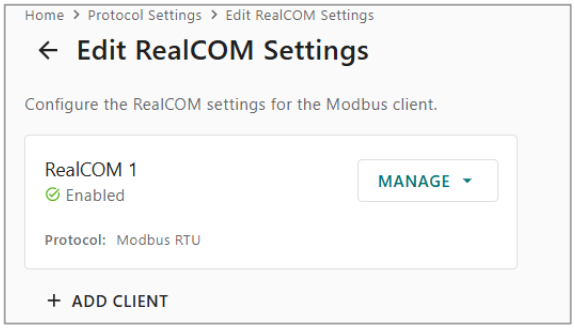
When MGate listens to the Modbus serial client, select **MANAGE**, and the following parameters can be changed according to your system requirements.

Setting	Value
Serial Port number	Serial port 1 (default), can be changed to other port numbers, depending on your MGate model
Protocol	Modbus RTU(default), Modbus ASCII
Enable/Disable serial port	Serial port is enabled by default. You can disable serial port(s) based on security requirements.

Modbus Real COM Client—MGate Real COM Settings

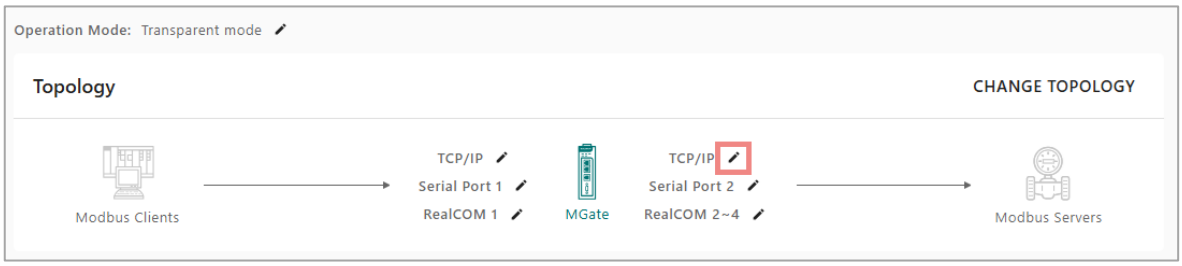


When the MGate listens to the Real COM client that runs Modbus serial programs, select **MANAGE**, and the following parameters can be changed according to your system requirements. Real COM is a virtual COM port of your computer that requires Windows Driver Manager (v4.4 or later) installation to activate. For more information about Real COM, check Real COM Mapping section.

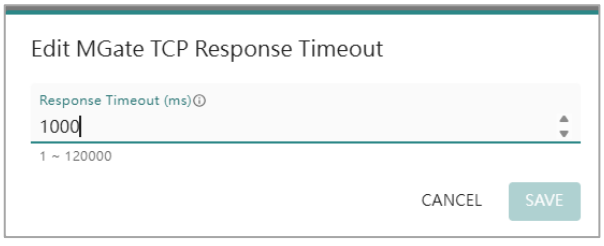


Setting	Value
Real COM Port number	Real COM Port 1 (default), can be changed to other Real COM port numbers: 2-4
Protocol	Modbus RTU(default), Modbus ASCII
Enable/Disable Real COM port	When activated in the topology, Real COM port is enabled. You can disable Real COM port(s) based on security requirements.

MGate Modbus TCP Settings—Modbus TCP Server

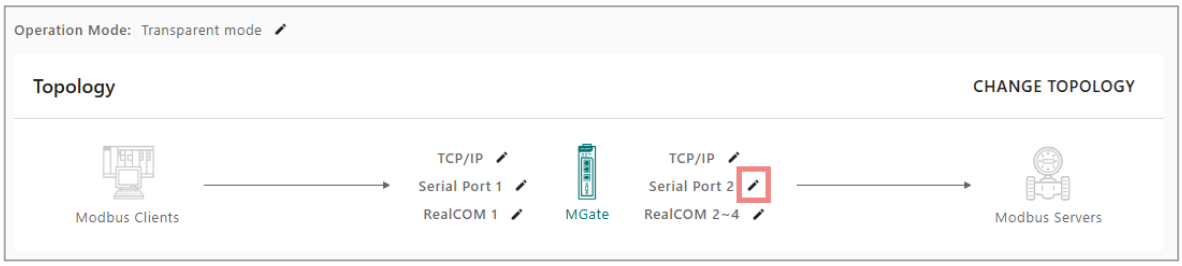


When the MGate connects to the Modbus serial server, the following parameters can be changed according to your system requirements.

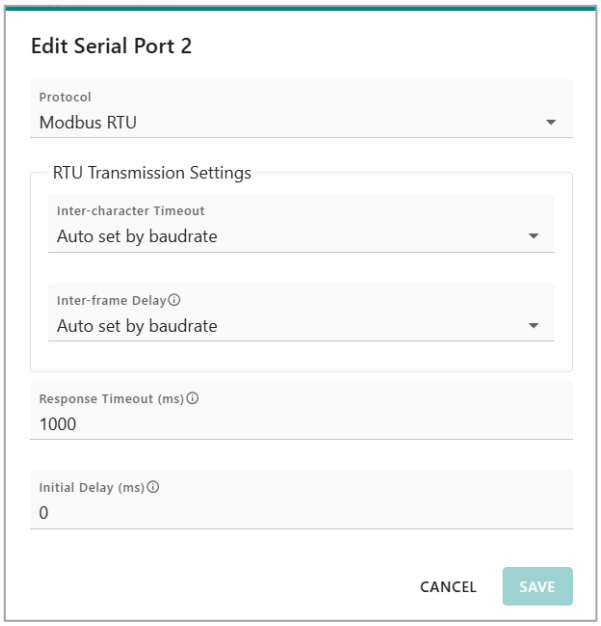


Setting	Value
MGate TCP Response Timeout (ms)	1000 ms (default), can be configured between 1~120,000 ms

MGate Modbus Serial Settings—Modbus Serial Server



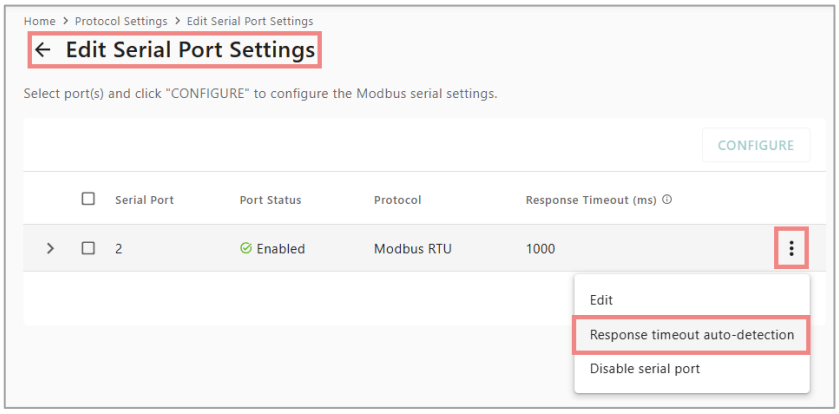
When the MGate connects to the Modbus serial server, the following parameters can be changed according to your system requirements.



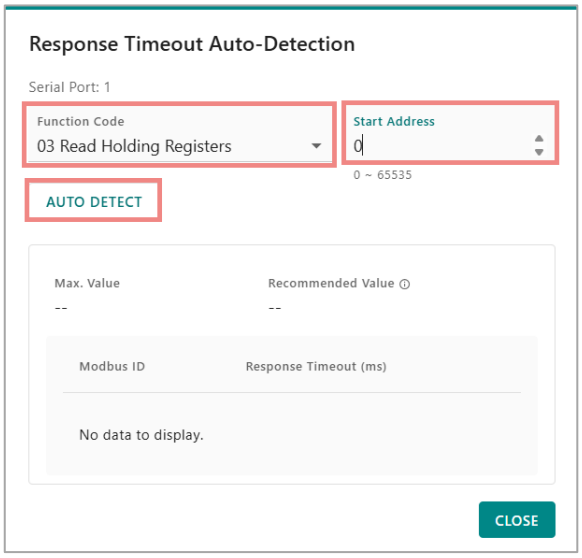
Setting	Value
Protocol	Modbus RTU(default), Modbus ASCII
Inter-character Time-out	Default: auto set by baudrate, can be user-fined between 10 to 500 ms
Inter-frame Delay	Default: auto set by baudrate, can be user-fined between 10 to 500 ms
Response Time-out (ms)	1000 ms (default), can be configured between 1 to 120,000 ms
Initial Delay(ms)	0 ms (default), can be user-defined between 0 to 30,000 ms. Some Modbus servers may take more time to boot up than other devices. For certain environments, this may cause the entire system to suffer from repeated exceptions during the initial boot-up. You can force the MGate to wait after booting up before sending the first request with the **Initial Delay** setting.
Enable/Disable serial port	Serial port is enabled by default. You can disable serial port(s) based on security requirements.

If your Modbus serial server keeps having response timeouts and you do not know how to adjust the settings, use **Auto Detection** to figure out the settings for you.

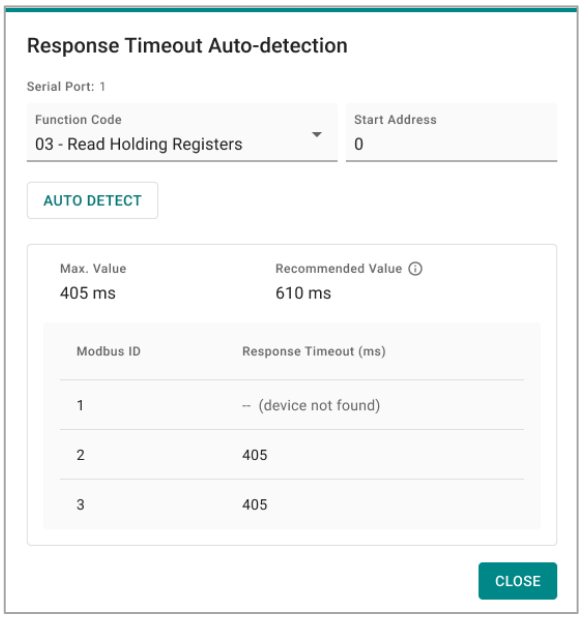
Auto Detection is a MGate feature that can help users find the appropriate Modbus response timeout for server devices. To enable this, first configure the Modbus routing rules for your serial ports, then select the pen icon in the topology, at last, select the edit icon and select "Response timeout auto-detection".



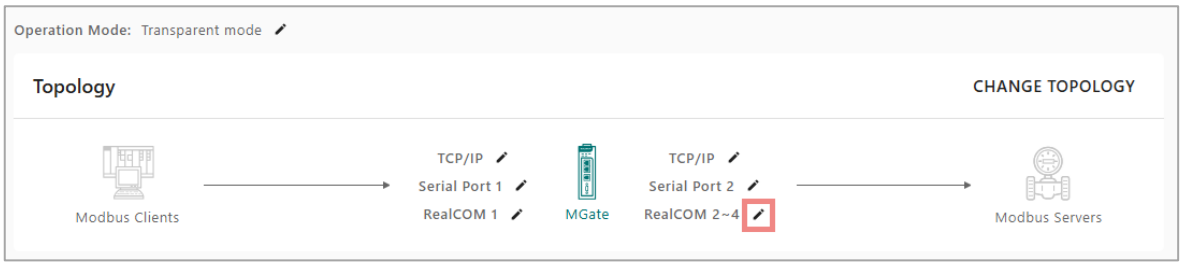
Select the function code to test for response timeout. Function codes 01, 02, 03, 04 are supported. Set the start address, and select **AUTO DETECT** to start the process.



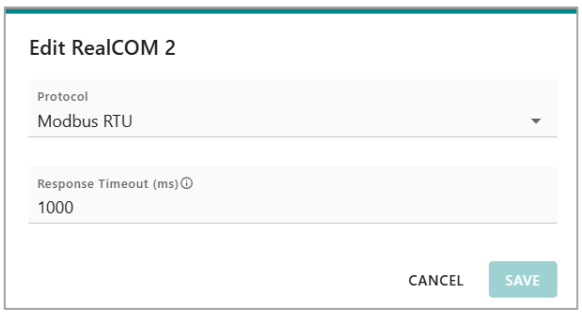
After the detection process, the MGate will give suggestions on the value of response timeout that you can configure. If MGate does not detect any responses from the target serial port, check your Modbus routing settings, the system topology, and if the end devices are responding.



MGate Real COM Settings—Modbus Serial Server



When MGate connects to the Modbus TCP server, the following parameters can be changed according to your system requirements.



Setting	Value
Protocol	Modbus RTU(default), Modbus ASCII
Response Time-out (ms)	1000 ms (default), can be configured between 1 to 120,000 ms
Initial Delay(ms)	0 ms (default), can be user-defined between 0 to 30,000 ms
Enable/Disable Real COM port	When activated in the topology, Real COM port is enabled. You can disable Real COM port(s) based on security requirements.

Modbus TCP Exception Explained

The MGate MB3000-G2 is a protocol gateway that transparently passes requests and responses between the Ethernet and serial interfaces. In some situations, it may be necessary for the gateway to return an exception in response to a request from a Modbus TCP client. This is enabled or disabled with the **Modbus TCP Exception** setting. When enabled, the unit can return three types of exceptions:

- Exception code 0x06: Server device busy (MGate's request queue is full) This exception code also equals MB_EXC_BUSY. In addition, up to 32 requests can be queued for each Modbus client.
- Exception code 0x0B: Gateway target device failed to respond (There is no response from the server. Maybe the device is offline, or the serial cable is broken.). Exception code also equals MB_EXC_NORESP.
- Exception code 0x0A: Gateway path unavailable (The destination ID is not included in the Modbus routing table, so the routing failed. The auto routing had ID conflict. The target Modbus TCP server is not connected or does not respond.). Exception code also equals MB_EXC_UNAVAILABLE.

Not all Modbus TCP clients require or support this exception. Configure this according to your requirements.

Modbus Response Time-out Explained

According to the Modbus standard, the time that it takes for a server device to respond to a request is defined by the device manufacturer (refer to Appendix A for details). Based on this response time, a client can be configured to wait a certain amount of time for a server's response. If no response is received within the specified time, the client will discard the request and continue operation. This allows the Modbus system to continue operation, even if a server device is disconnected or faulty.

On the MGate MB3000-G2, the **Response Time-out** field is used to configure how long the gateway will wait for a response from a Modbus server. Refer to your device manufacturer's documentation to manually set the response timeout. The MGate MB3000-G2 also provides an automatic calibration of the response time-out. Instead of manually figuring out the appropriate setting, you can select **Auto Detection** to have the MGate figure out the Modbus serial response timeout settings for you. Once a value has been recommended, you can fine-tune it for the best performance.

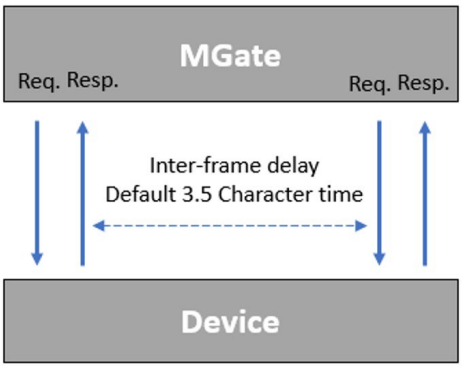
Inter-character Time-out Explained

When connecting to Modbus RTU server, use this function to define the time-out interval between characters in one frame. When the serial side of MGate receives one character, and the next one comes after the "inter- character timeout" defined, the frame will be discarded because of time-out. The inter-character timeout can be adjusted within the range of 10 to 500 ms or use the default value. The default value in this field is 0 ms, but the default inter-character timeout varies, depending on the baudrate setting. When the baudrate is configured below 19200 bps, the default inter-character time-out is set to 1.5-character times. When the baudrate is configured equal to or larger than 19200 bps, a predefined fixed value is used.

Inter-frame Delay Explained

When connecting to Modbus RTU server, use this function to define the time interval between a Modbus RTU response and the next Modbus RTU request. The reason for enabling manual configuration of this value is to accommodate certain scenarios where legacy Modbus devices may not process Modbus RTU requests rapidly. By setting a longer interval value, you can increase tolerance for delays in serial reception and transmission. The inter-frame delay can be adjusted within the range of 10 to 500 ms or use the default value. The default value in this field is 0 ms, but the default inter-frame delay varies depending on the baudrate setting. When the baudrate is configured below 19200 bps, the default inter-frame delay is set to 3.5-character times. When the baudrate is configured equal to or larger than 19200 bps, a predefined fixed value is used.

How do you calculate Modbus character time? For example, if the baudrate is 9600bps, then 1 character time is about 1ms. In a serial frame (11bits, including start bit, data, parity bit and stop bit), 9,600bps approximately equals 960 characters/s, so transmitting one character needs about 1/960=1ms.



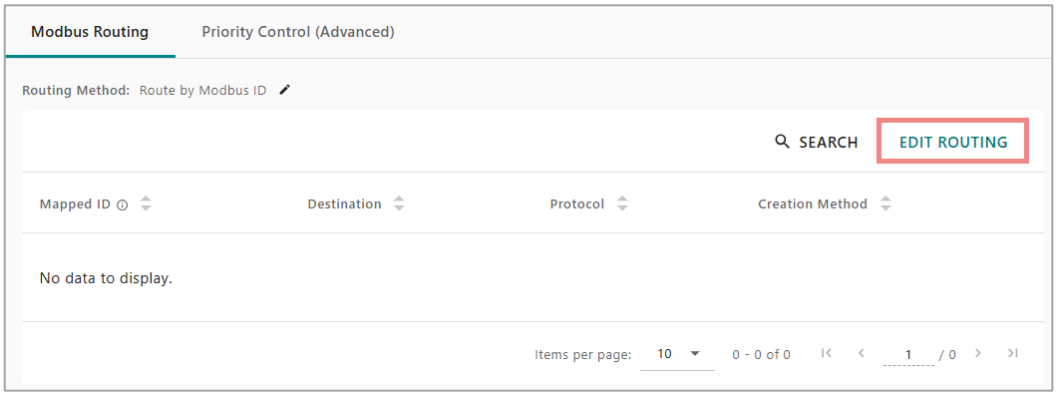
Modbus Routing

Modbus Routing is a mechanism for gateways to manage Modbus requests and route them to the specific serial ports that connect the targeted Modbus server devices, or to specific TCP devices. This keeps communication efficient and prevents devices from other serial ports from receiving an unrelated Modbus requests, resulting in slowing down the entire system. There are three types of Modbus routing mechanisms. The default is **Route by Modbus ID**, which decides how Modbus IDs are routed to which serial port.

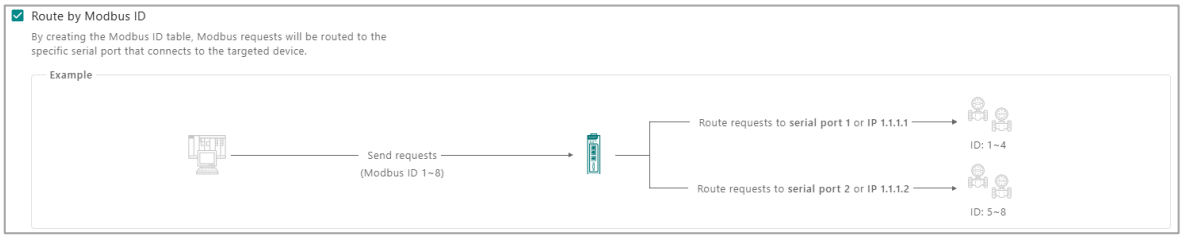
You can also enable other Modbus routing mechanisms: **Route by IP address**, or **Route by TCP port**. You can also enable multiple Modbus routing methods at the same time. The details will be explained in the following section.

First, we will introduce the default routing method: **Route by Modbus ID**.

Select the **EDIT ROUTING** button to add a Modbus routing rule.

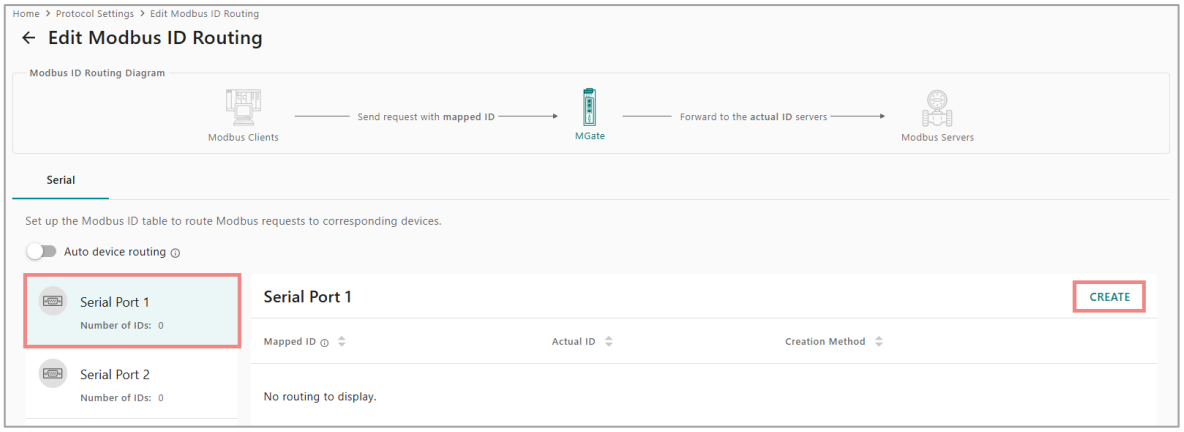


The settings on this page determine how Modbus requests will be routed by the MGate. Since the Modbus devices (all with different Modbus IDs) are connected to the different serial ports of a gateway, the Modbus requests should be routed to the specific serial port that is connected to the targeted Modbus server device. For example, Modbus ID 1, 2 is connected to serial port 1, and Modbus ID 5, 6 is connected to serial port 2, by settings up routing, the MGate will 'route' requests to Modbus ID 1, 2 to serial port 1, and route requests to Modbus ID 5, 6 to serial port 2. If you are using a system that connects Modbus TCP server devices, you can also configure Modbus routing to route to specific IP addresses.

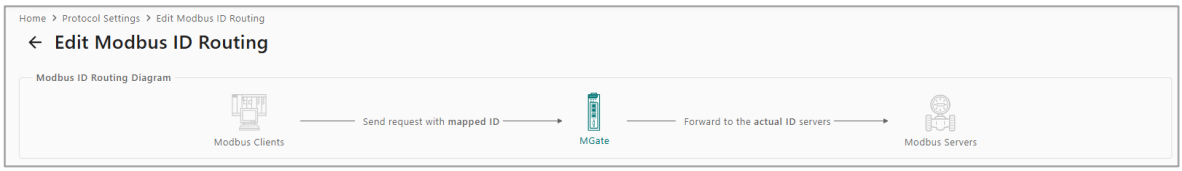


For 1 serial port model, there is a factory default Modbus routing. Since there is only one port, we don't have to specify other serial ports, so all the Modbus requests (requests with Modbus IDs 001 to 255) will be routed to serial port1. For models with more than 1 port, there is no default routing so users must configure their own routing rules according to the system topology.

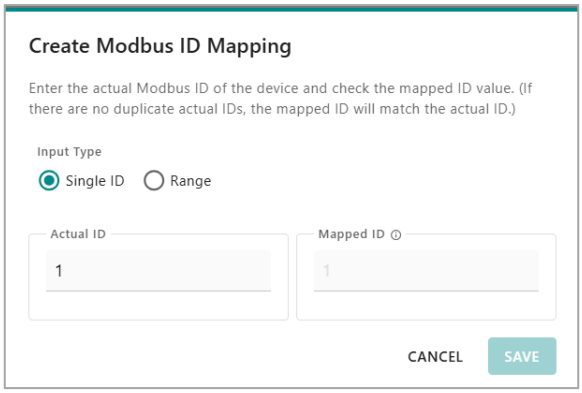
To configure your own routing, after selecting **EDIT ROUTING**, select the desired serial port, and select **CREATE**.



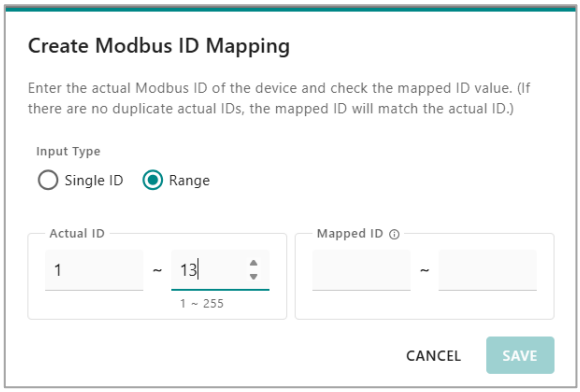
There are two kinds of methods, you can configure single Modbus ID or configure a range of IDs. Also, you will have to configure the Actual ID and Mapped ID. The Actual ID is the real Mobus ID of your end devices. The Mapped ID is the Mobus ID requested from the Modbus client.



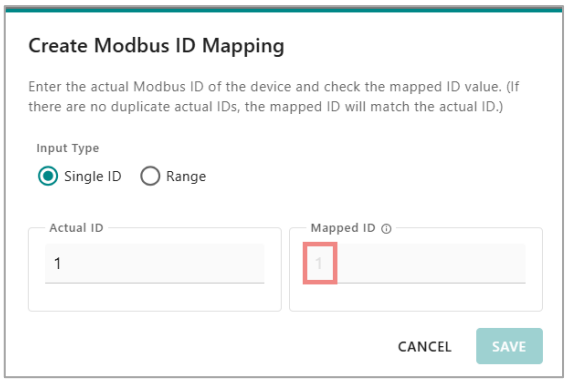
Example of configuring a single ID:



Example of configuring a range of IDs:



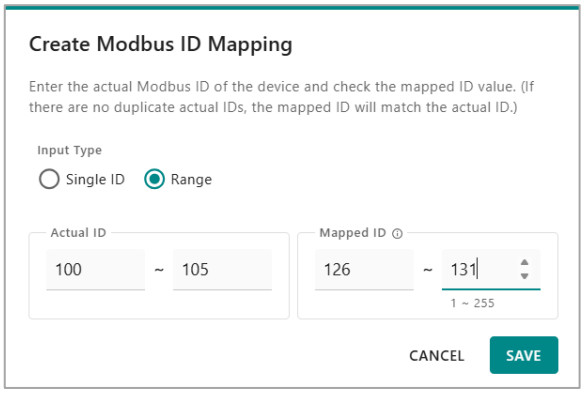
Most times, the Modbus client requests the end device's ID directly without changing, so the Actual ID is the same as the Mapped ID, and the web console will guide you to configure the same ID(s).



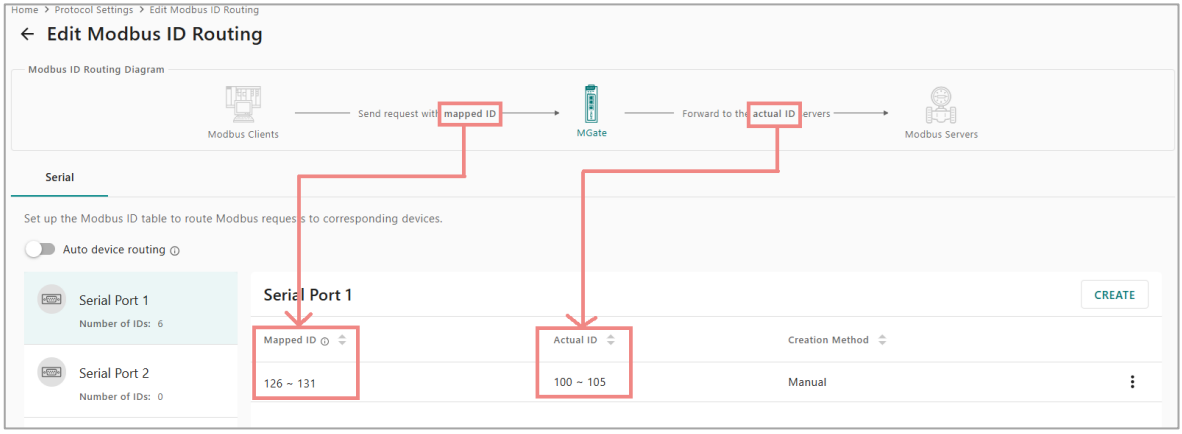
However, sometimes, the Modbus IDs for legacy Modbus devices cannot be changed. They have a fixed ID or a range of Modbus IDs. Connecting the same Modbus IDs to different serial ports of MGate will cause a conflict with the Modbus protocol rule. The Mapped ID function can help you to connect the same Modbus ID devices to different serial ports of a protocol gateway. Let's assume there are two legacy Modbus devices (named Device A, and Device B) using the same Modbus ID 1. Device A is connected to serial port 1 and Device B is connected to serial port 2. On the Modbus Client side, you can configure the Modbus request for Device A is recognized by Modbus ID 1 and the request for Device B is recognized by Modbus ID 2. Now, the Modbus client can send a request with Modbus ID 1 to Device A connected to serial port 1, and send a request with Modbus ID 2 to Device B connected to serial port 2.

	Mapped ID (Request from Modbus client)	Actual ID (The ID of the end device)
Device A	1	1 Note: This device is connected to serial port 1.
Device B	2	1 Note: This device is connected to serial port 2.

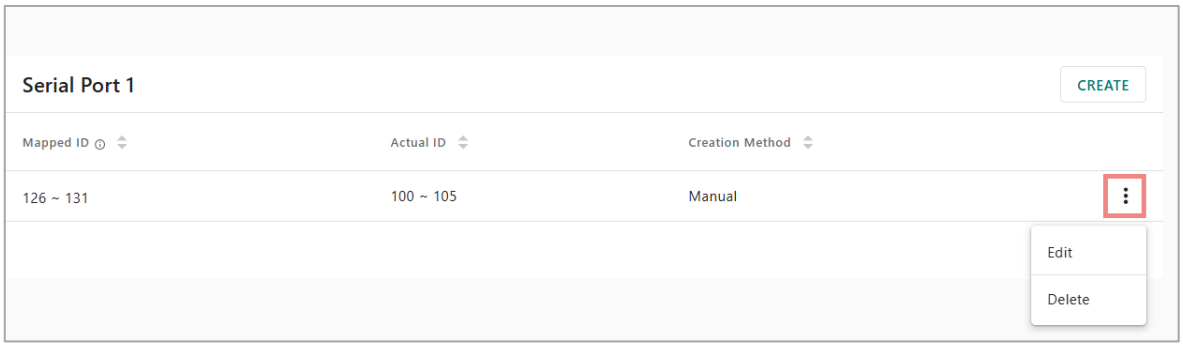
Example of a configured mapped ID:



After configuring Modbus ID mapping, view the results below. The Mapped ID represents the ID recognized by the Modbus client and can be different from the actual Modbus ID of end devices.

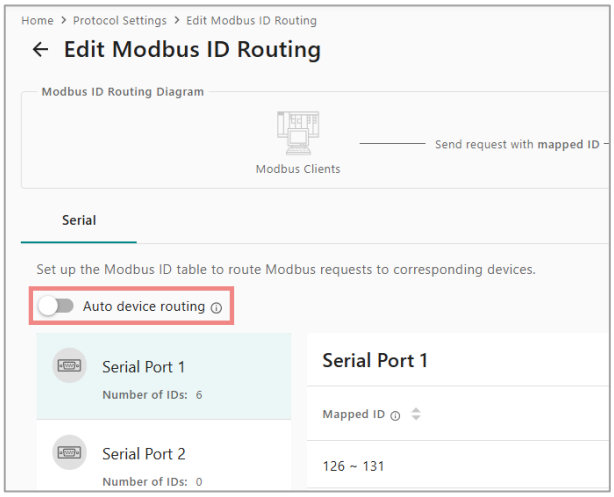


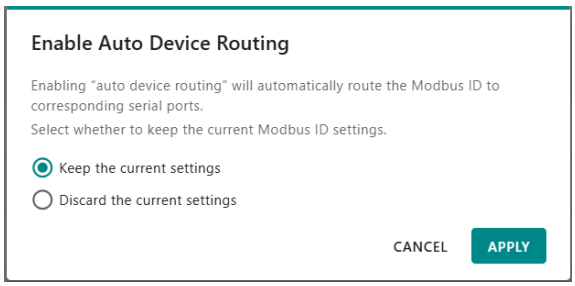
If you want to change or delete the configured Modbus routing, select the edit button besides the routing rules.



Auto Device Routing

The Moxa Modbus gateways provide an automatic routing mechanism that eliminates the burdensome task of setting the Modbus IDs table manually. By using this function, you no longer need to set the routing table. The Moxa Modbus gateways will help you detect the Modbus serial devices connected and route correctly. Use the switch bar to enable Auto Device Routing, and a message window will pop up.





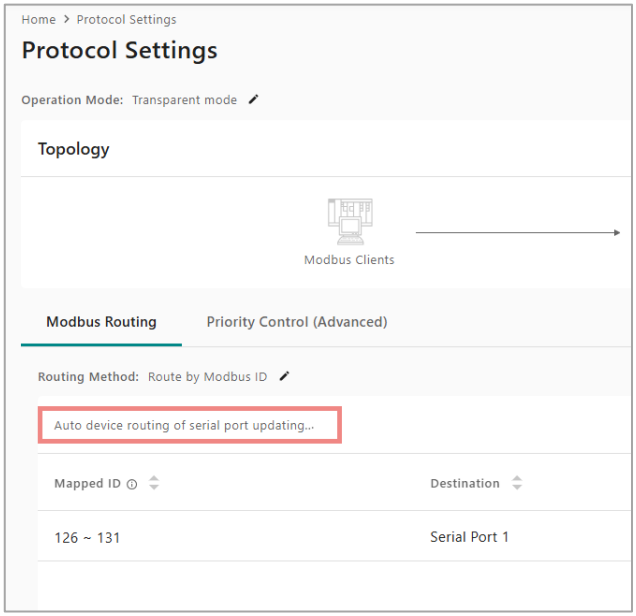
Select **Discard the current settings** to delete the existing routing table. The auto-routing mechanism will automatically find the correct serial port that connects the target Modbus device. Moreover, if a device is added to the gateway later, the gateway can also route it correctly.

If you have manually set the routing table and would like to enable the auto-routing mechanism for the newly added devices, select **Keep the current settings** to keep the existing routing table. The MGate will retain the existing user-set routing table and automatically detect newly added devices.

NOTE

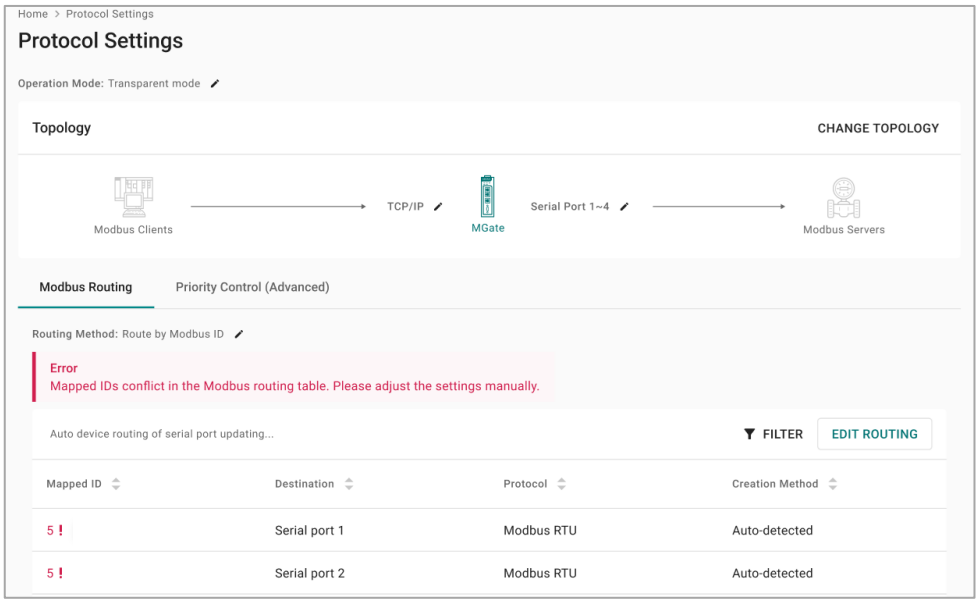
If a newly added device cannot be polled by the Modbus client correctly, the Modbus IDs of this newly added device might be set in the existing user-set table. See how to solve the ID conflict issue below.

If the Auto Device Routing is enabled, you can see the status on the Protocol Settings page, as below:

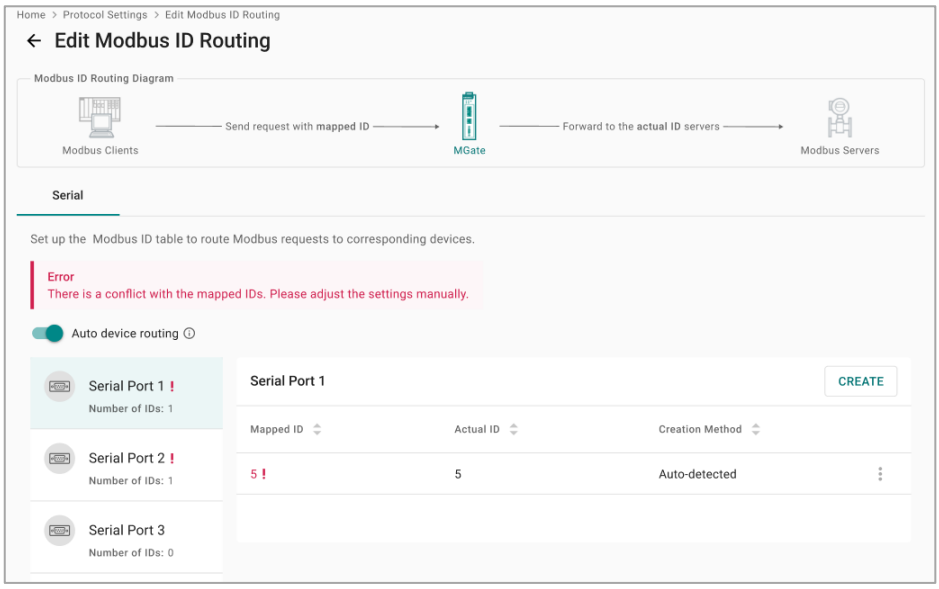


The Auto Device Routing rules cannot use the Mapped ID feature, since the gateway will not know what mapped IDs are the Modbus client going to use. So, when using Auto Device Routing, the Mapped ID will always be the same as Actual ID.

If a conflict exists, for example, two Modbus devices with the same Modbus IDs are connected to serial port 1 and port 2. The table will show an exclamation mark "!" and the conflict Modbus ID and error message in red.

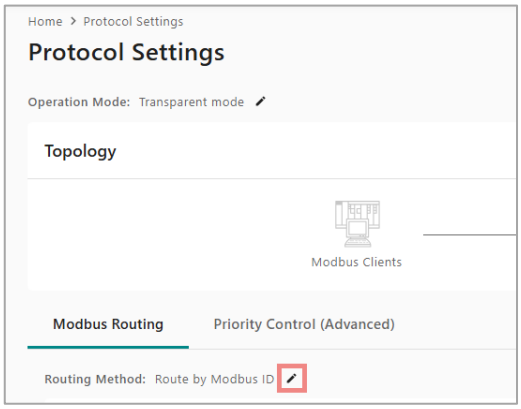


To check which serial port detected the conflict, select **EDIT ROUTING**, then you can see which serial ports have the exclamation mark "!" and error message in red.



How to solve the ID conflict issue: Change the end device's Modbus ID, or configure the Modbus routing manually with mapped ID instead of using auto device routing.

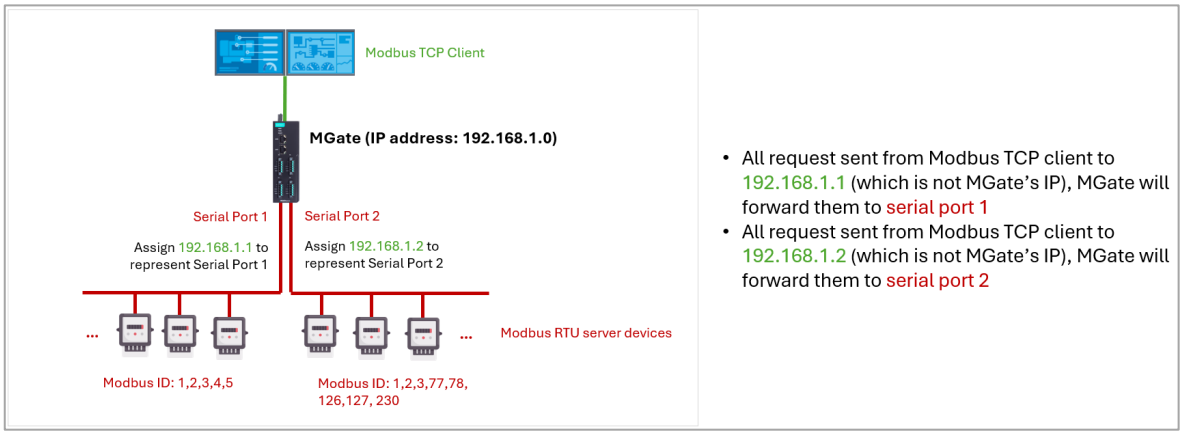
Other than the default **Route by Modbus ID**, to enable **Route by IP address** or **Route by TCP port**, select the edit button besides Routing Method, and use the checkbox to select desired routing methods.



There are some scenarios suitable for using **Route by IP address** and **Route by TCP port**:

1. When you have multiple Modbus serial devices that use same Modbus ID, and the system architecture and Modbus ID deployment is very complicated, other than using Route by Modbus ID, you can also use **Route by IP address** or **Route by TCP port**.
2. When your Modbus system design requires you to map every MGate's serial port to an IP address or to a TCP port.

The Modbus client can communicate with the Modbus server devices connected to a specific serial port on the gateway by assigning an IP address or TCP port to a specified serial port. Under these two kinds of routing mechanisms, each serial port can be accessed by up to four Modbus clients.

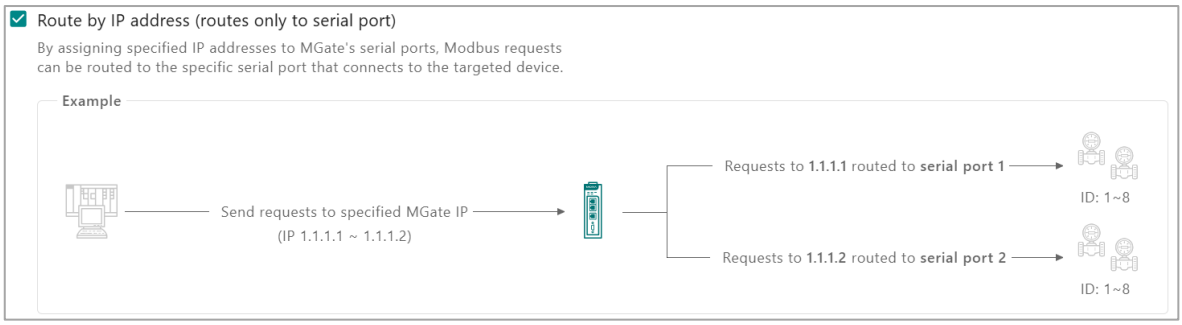


Using **Route by IP address**, for example, you can assign IP address 192.168.1.1 to serial port 1 and assign IP address 192.168.1.2 to serial port 2. When the gateway receives a Modbus request sent to 192.168.1.1, the gateway will forward the Modbus request to serial port 1 directly, and when the gateway receives a request to 192.168.1.2, it will forward the Modbus request to serial port 2.

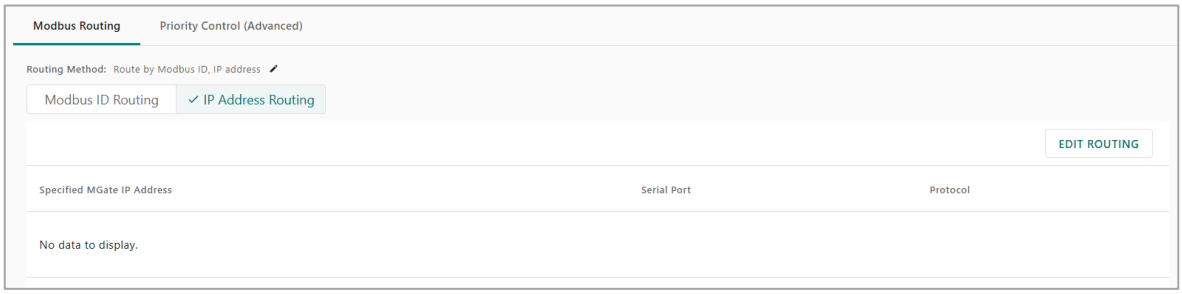
NOTE

These IP addresses need to be set different to MGate's own IP address. These IP addresses need to be using the MGate's Modbus TCP listen port, which is usually 502, or check the MGate's listen port settings.

To configure the rules, select Route by IP address and apply.



You will see the **IP ADDRESS ROUTING** tab appear. Select **EDIT ROUTING** to add routing rules.

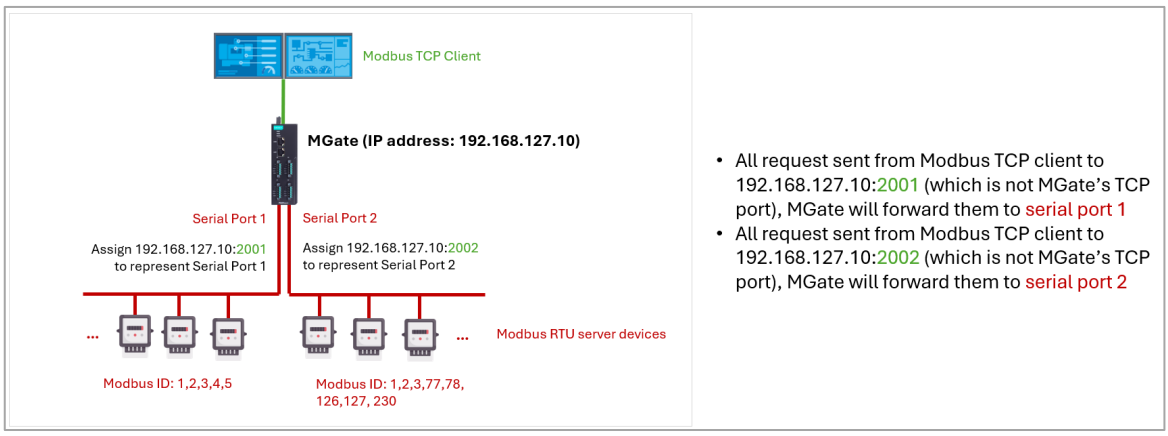


On this page, you can assign the IP addresses to specific serial ports. One IP address maps to one serial port. Select SAVE to take effect.

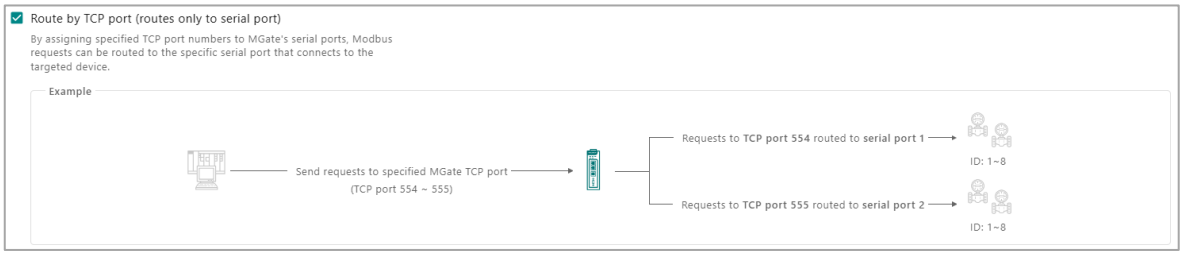
Using **Route by TCP port**, for example, you could assign TCP ports 2001 to 2002 to serial ports 1 and 2 respectively. When the gateway receives a Modbus request for TCP port 2001, it will forward the Modbus request to serial port 1, and it does not care what IDs are under serial port 1. Similarly, if it receives a Modbus request for TCP port 2002, it will forward the Modbus request to the serial port 2 directly.

NOTE

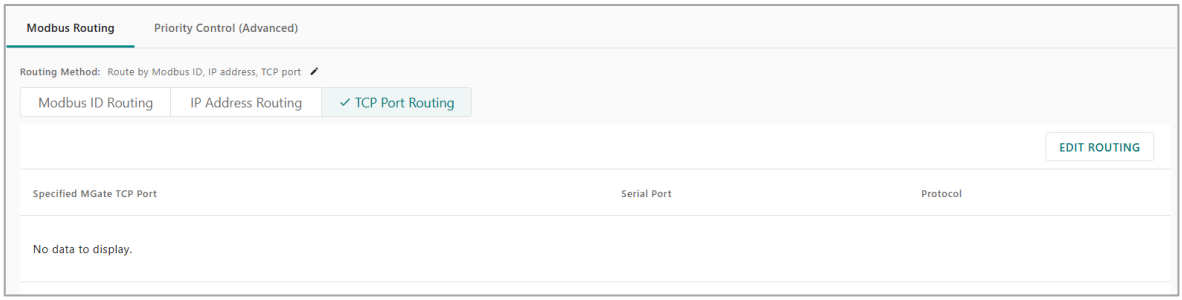
The TCP ports need to differ from MGate's own Modbus TCP listen port (usually TCP port 502). The TCP ports need to use the MGate's IP address.



To configure the rules, select Route by TCP Port and apply.



You will see the **TCP Port ROUTING** tab appear. Select **EDIT ROUTING** to add routing rules.

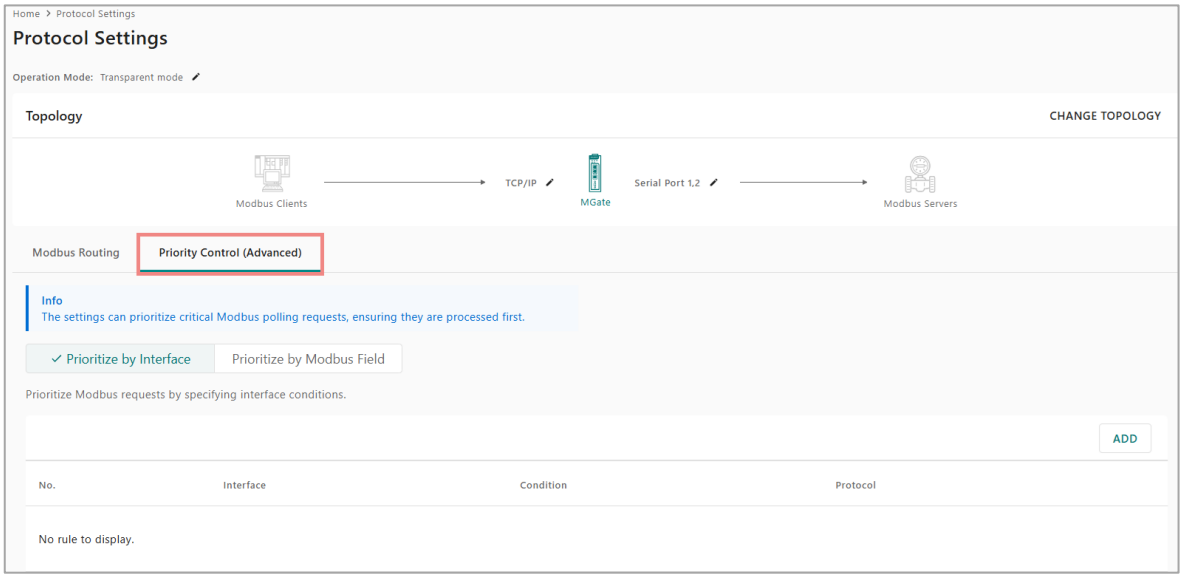


On this page, you can assign the TCP ports to specific serial ports. One TCP port maps to one serial port. Select SAVE to take effect.

With **Route by IP address** and **Route by TCP port** routing mechanisms, Modbus ID routing can be simpler and more effective by giving you the flexibility to define a multirouting mechanism.

Priority Control

The Priority Control tab is where emergency requests are enabled and configured. This is available for advanced models (MGate MB3x70-G2 series) only.



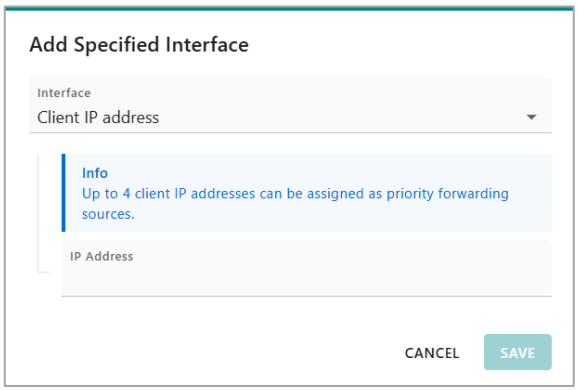
Priority control is designed for requests that are sent to Modbus RTU/ASCII servers. Since Modbus RTU/ASCII servers cannot handle multiple requests, the Modbus gateway must send each request individually and wait for the response before sending the next request. As requests stack up, the response time can be long. This can cause problems for certain critical requests that require an immediate response.

With priority control, you can specify the Modbus requests to be inserted to the front of the waiting queue for more immediate response times. There are two methods to prioritize your Modbus requests: **Prioritize by Interface** or **Prioritize by Modbus Field**.

Prioritize the Modbus requests **by Interface**, using the following three conditions:

1. **Client IP address:** Specify which Modbus client's IP address needs higher priority (up to 4 Modbus client's IP address)
2. **MGate Serial port:** Specify which serial port connected to Modbus client needs higher priority (need to enable Modbus serial client in the TOPOLOGY)
3. **MGate TCP port:** Specify which MGate listen TCP port needs higher priority (you need to specify a TCP listen port other than the Modbus listen port, which is usually 502. Specify ports between 1~65535)

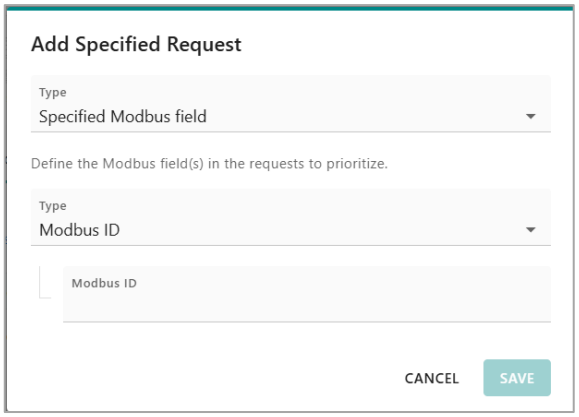
When the Modbus gateway identifies a priority request, the request will immediately be placed at the front of the queue. Select the **Prioritize by Interface** tab, select the **ADD** button to add Modbus requests that need higher priority.



Select the condition that you want to configure and fill in the related settings.

For example, if you want all requests from Modbus TCP client 192.168.32.161 to be considered a priority request, do the following:

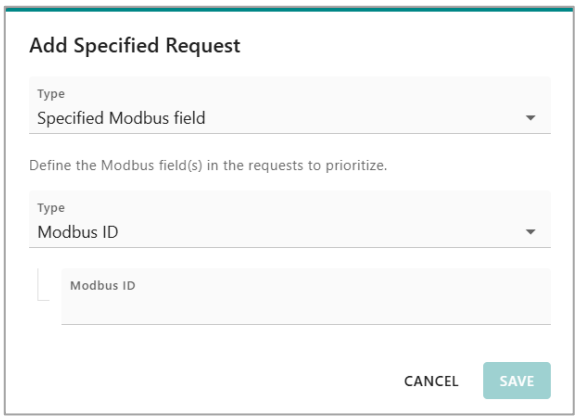
1. Select **Client IP address** from the drop-down menu
2. Enter 192.168.32.161
3. Select SAVE



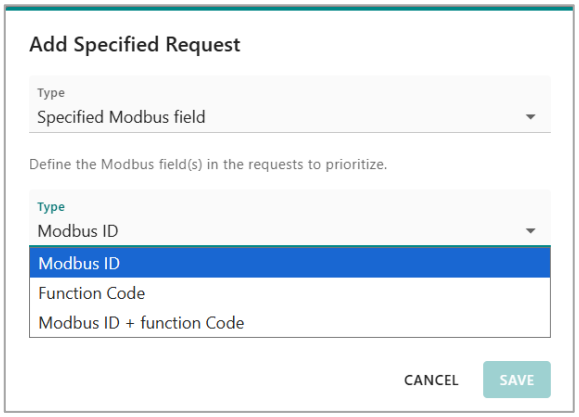
You can also prioritize the Modbus requests by defining the Modbus command fields, using the following conditions:

1. **Specified Modbus field > Modbus ID:** Specify the Modbus ID that needs higher priority.
2. **Specified Modbus field > Function Code:** Specify the Function Code that needs higher priority.
3. **Specified Modbus field > Modbus ID + Function Code:** Specify the Modbus ID +Function Code that needs higher priority.
4. **User-defined frame:** Define a full Modbus frame that needs higher priority by using Hex.

When the Modbus gateway identifies a priority request, the request will immediately be placed at the front of the queue. Select the **Prioritize by Modbus Field** tab and the **ADD** button to add Modbus requests that need higher priority.



Select the type that you want to configure and fill in the related settings.



For example, if you want all requests with Modbus ID 6 to be considered a priority request, do the following:

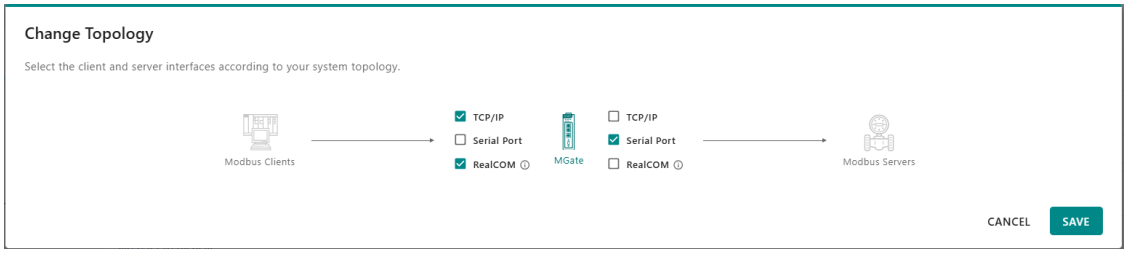
1. Select **Specified Modbus field** from the drop-down menu
2. Select **Modbus ID** as the Modbus field type
3. Fill in Modbus ID **6**
4. Select SAVE

Real COM mapping

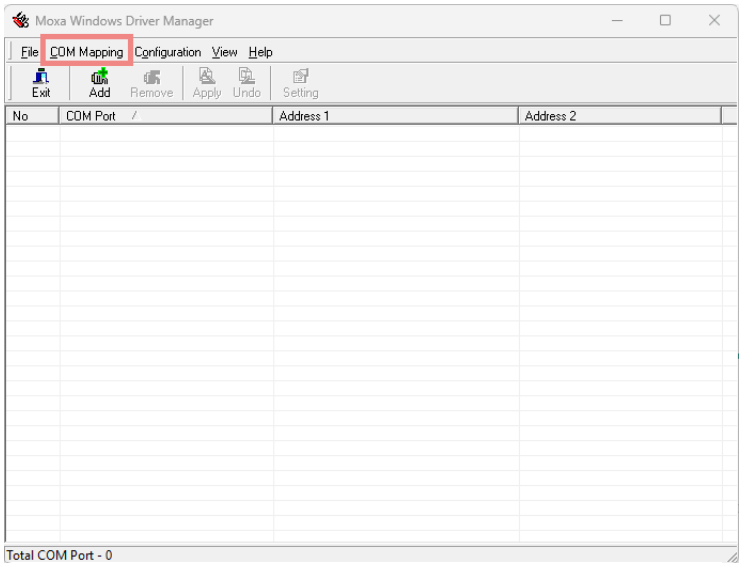
If your system uses PCs that can only run COM port-based Modbus programs to monitor or control Modbus devices, then the Real COM function is the best solution for your system. By using Real COM, we help create a virtual COM port on your PC, and it acts as a real COM port over an Ethernet network. The MGate MB3x70-G2 Real COM will treat your PC's COM port as if it were an additional serial port on the MGate itself.

Configuring Real COM

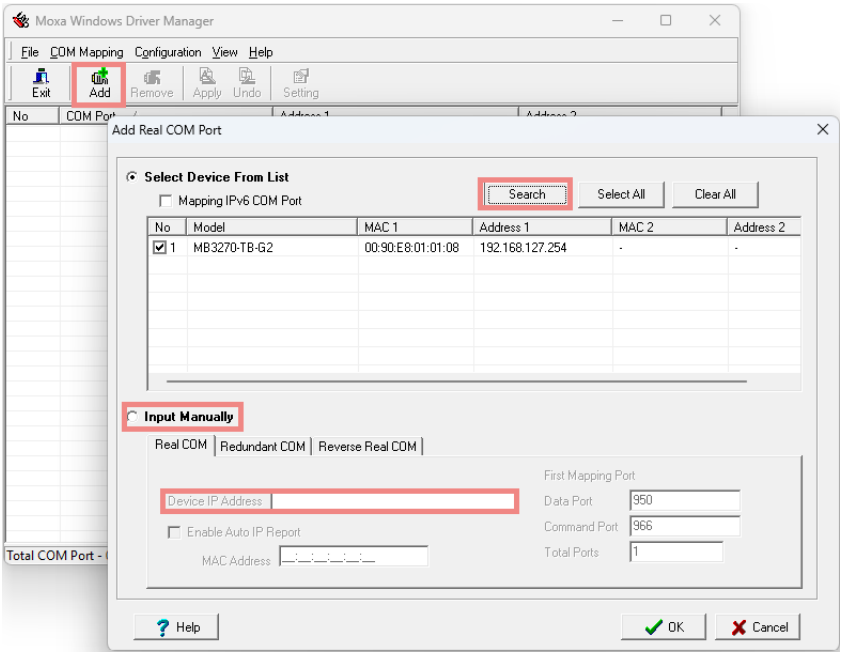
1. Enable Real COM. Enable Real COM on the MGate's web console by navigating to Protocol settings, select the **CHANGE TOPOLOGY** button, and select Real COM.

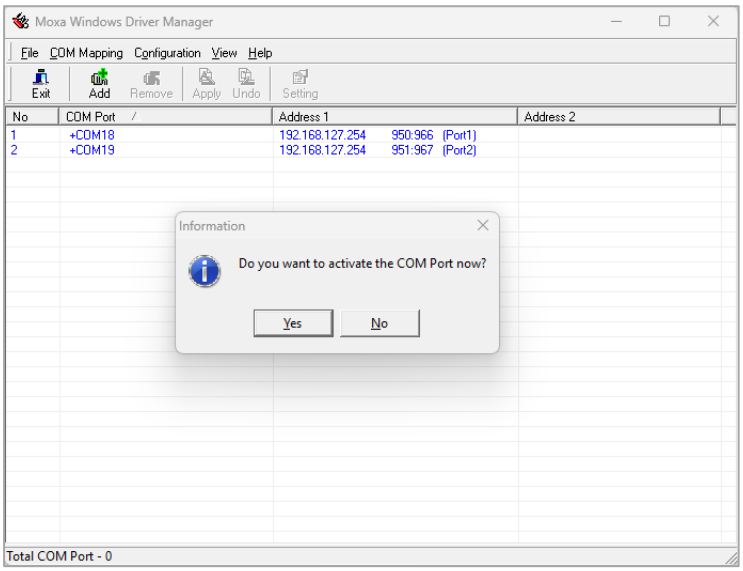


2. Add the MGate device to the driver: Download Windows Driver Manager v4.4 or later versions. Download Windows Driver Manager from the Moxa website, select **COM Mapping > Add** to add the MGate device. Use **Search** to find the MGate IP, or use **Input Manually** to fill in MGate's IP address, and select **OK** to take effect.



3. Map the virtual COM port. After adding the MGate to Windows Driver Manager, it will pop-up a message to activate the COM port. Select **Yes** to take effect.



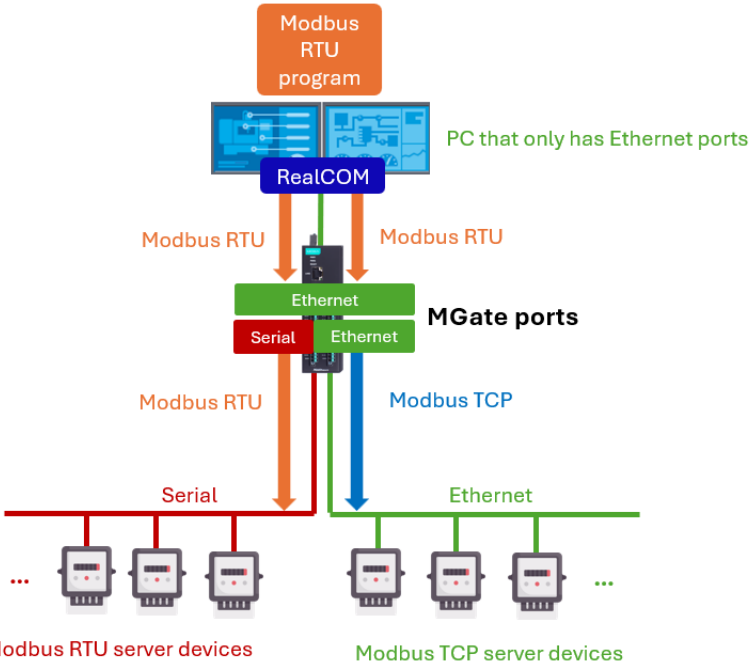


Now you can see the mapped virtual COM port in Windows Driver Manager. Use these COM port numbers with your COM-based program to communicate over Ethernet.

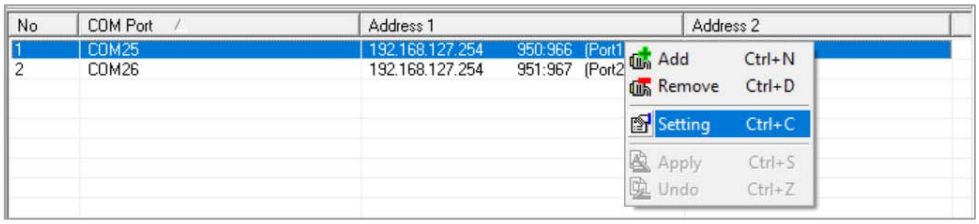
No	COM Port	Address 1		
1	COM25	192.168.127.254	950:966	(Port1)
2	COM26	192.168.127.254	951:967	(Port2)

Map up to four Real COM ports for each Modbus gateway to your PC's COM ports. The driver will generate virtual COM ports on your PC to connect to the selected MGate MB3x70-G2 over the network.

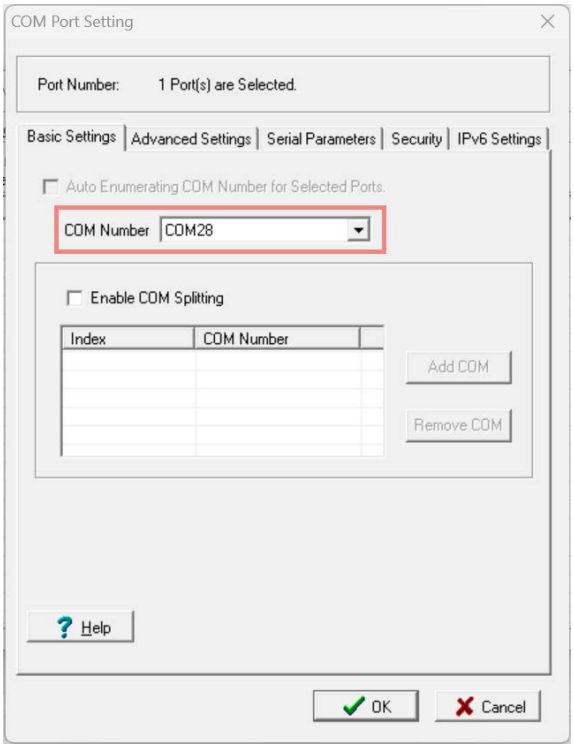
This way, when you send a Modbus request to Real COM port, the driver will forward your request to the MGate MB3x70-G2 and the MGate MB3x70-G2 will forward the request to the target Modbus server device using the Modbus routing table. For example, the Modbus request from Real COM can be redirected to a Modbus RTU/ASCII server device that is connected to the MGate MB3x70-G2's serial port, or to a Modbus TCP server device through the MGate MB3x70-G2's Ethernet port. In addition, it can be redirected to another Real COM port on the MGate MB3x70-G2.



If you would like to change the Real COM port number, right-select or double-select the mapping results on the Windows Driver Manager.

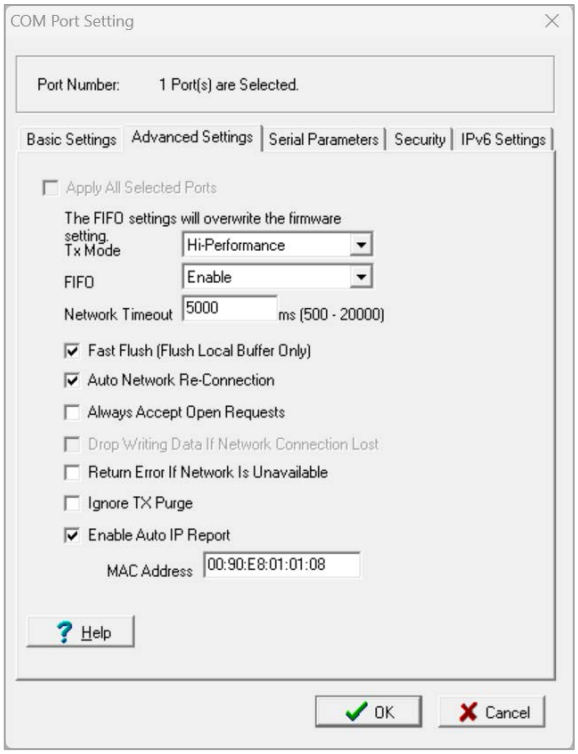


Use the drop-down menu to change the COM port number, select **OK** and check if the COM port is changed. successfully.



Always Accept Open Requests

This function enables users to open Real COM port(s) before the MGate device is connected. Here, the data transmitted to Real COM port(s) will be stored temporarily and will be sent out once the Real COM port(s) are ready for access. Right- select or double select the mapping results on the Windows Driver Manager, go to the **Advanced Settings** tab, check the **Always Accept Open Requests** checkbox and select **OK** to enable this function.



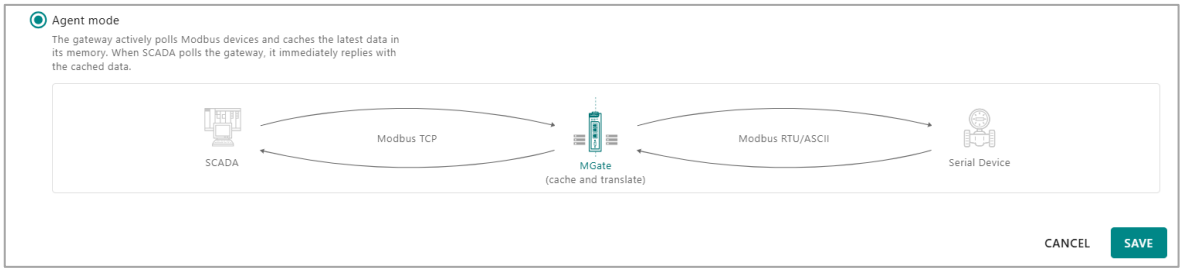
Agent Mode

In **Transparent Mode**, the MGate works like a "pass-through" device. The gateway simply converts messages between Modbus TCP (Ethernet) and Modbus RTU/ASCII (serial). Only one device can talk at a time, so other devices must wait their turn. The advantage is it is easy to use with existing SCADA systems. However, the drawback is slower performance, especially when many Modbus serial devices are connected.

In **Agent Mode**, the gateway has an **active role**. Instead of waiting for SCADA to ask for data, the gateway actively and regularly collects data from multiple devices at the same time and **caches it in its own memory**. In this case, SCADA systems can then quickly retrieve this stored data in MGate without waiting for end devices to respond, **greatly improving performance**. The drawback is that users must manually configure all the Modbus commands that they need the MGate to actively poll.

NOTE

Switching the operation mode will clear all protocol settings and will require reconfiguration.

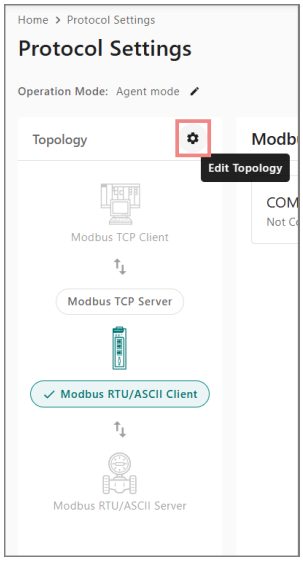


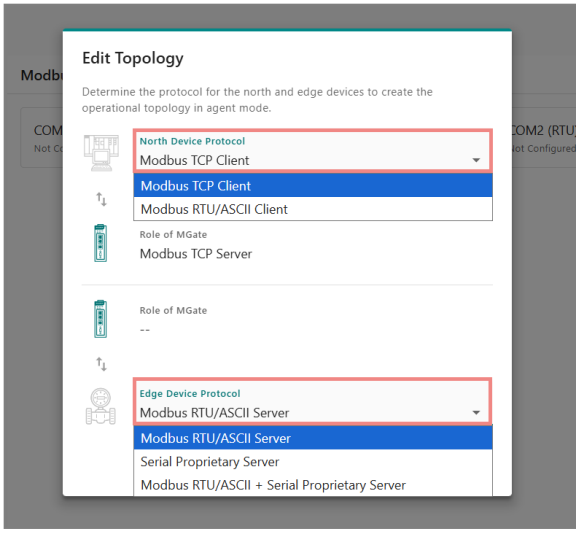
The default topology is the most seen Modbus TCP client—Modbus serial server (RTU).

To change the topology, select the settings button beside topology and configure it according to your application topology.

From the device's perspective, the MGate MB3x70-G2 models support the following topologies:

- Converting Modbus serial server device to Modbus TCP client device
- Converting Modbus TCP server device to Modbus serial client device
- Converting serial proprietary server device to Modbus TCP client device
- Converting Modbus serial server device + serial proprietary server device to Modbus TCP client (only on multi-serial port model) device





After selecting the conversion topology you need, select **SAVE** for the topology to take effect.

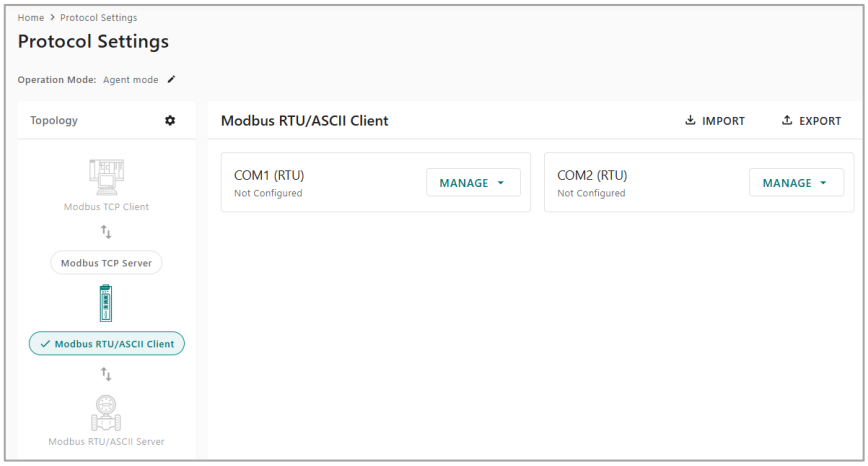
The topology will show the respective MGate role. To configure MGate's Agent Mode protocol settings, just select the MGate role.

In the following sections, we will introduce the configuration of every MGate's protocol role.

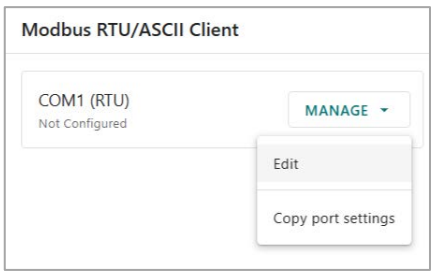
We suggest configuring in this direction: **Protocol A Client settings, set commands > Protocol B Server settings, add tags and mapping from protocol A to B**.

Modbus RTU/ASCII Client

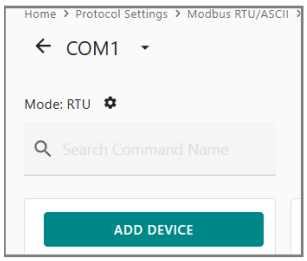
By selecting on the MGate Modbus RTU/ASCII client role, you can configure the Modbus commands that MGate will actively poll. The MGate supports csv file import/export for the full Modbus client settings, including command settings; it is easy to use when you back up the settings or offline configuration during the installation stage. We suggest exporting the file first and then importing it back to the MGate after configuring it.

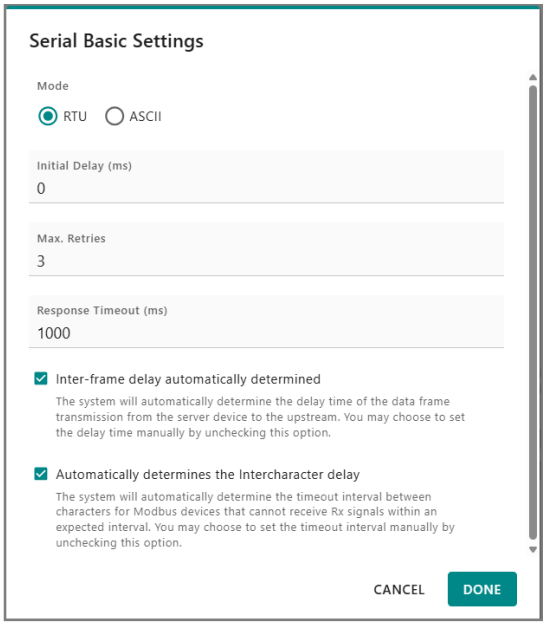


Choose the desired serial port, select the **MANAGE** button and **Edit**. If you need to copy the same settings to other seral port(s), select Copy port settings and the target port(s).

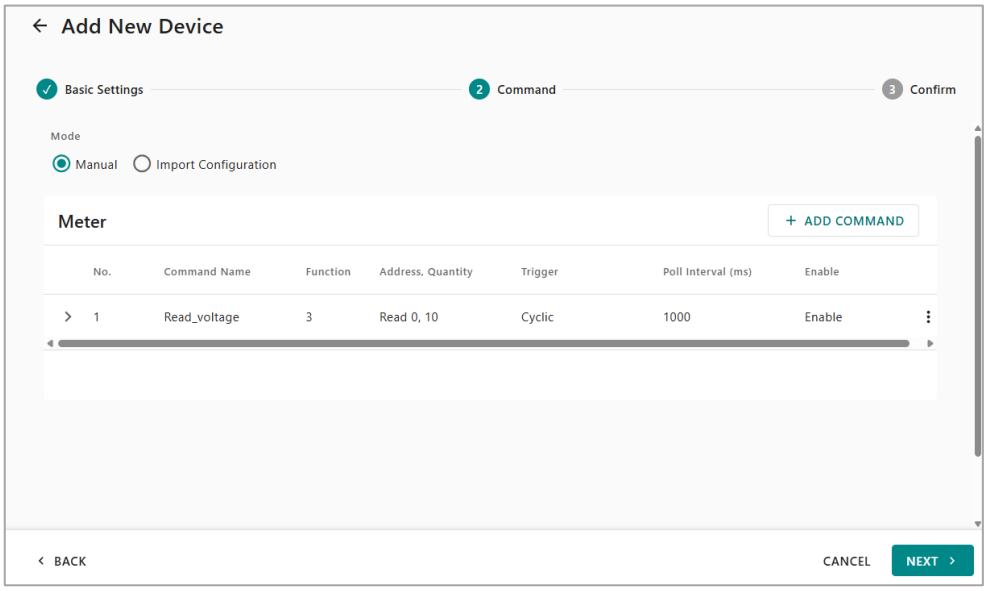


The default protocol is Modbus RTU. To change to Modbus ASCII, select the button besides the protocol, and change the related Modbus settings.

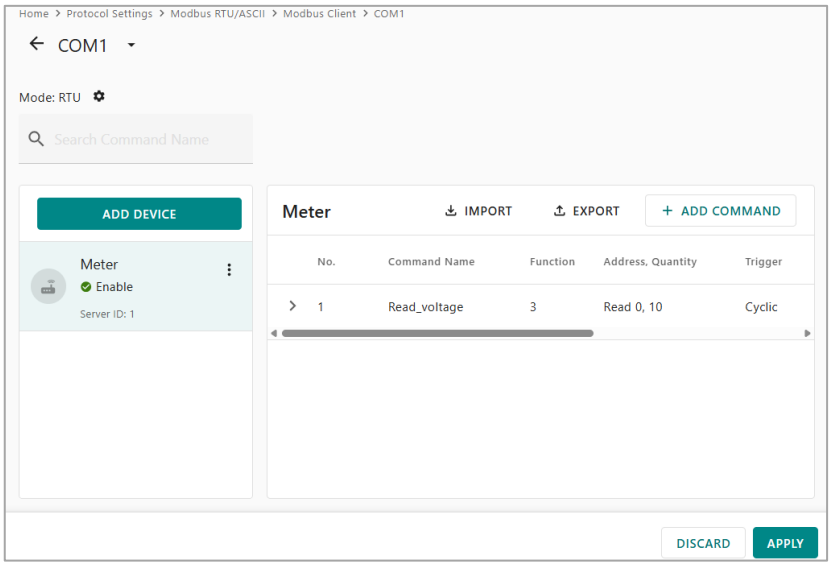




Add a Modbus device by selecting **ADD DEVICE**, set up the name and the target Modbus server ID that you are going to ask, and select **ADD COMMAND** to add the Modbus commands. The MGate supports csv file import/export for Modbus command settings. We suggest exporting the file first and then importing it back to the MGate after configuring it.



Confirm your settings and select **DONE**. Finally, remember to select **APPLY** to take effect.



When a message is sent from a client to a server device, the function code field tells the server what kind of action to perform.

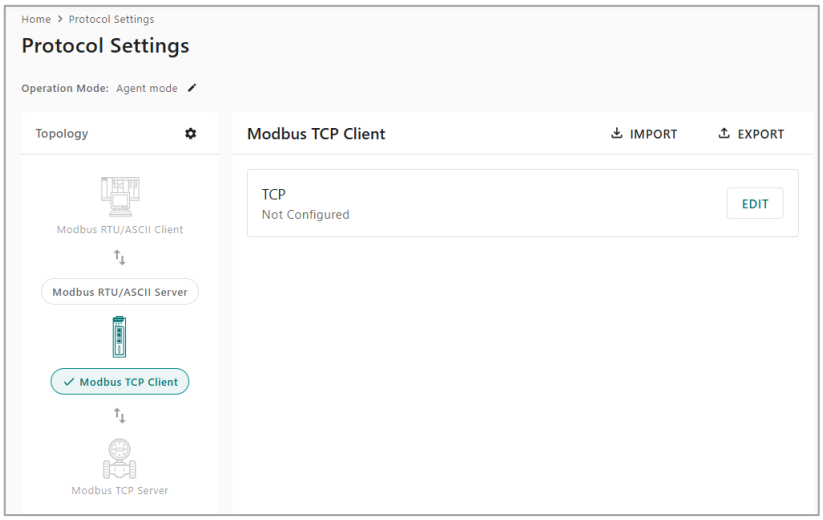
Parameter	Value	Default	Description
Command Name	Alphanumeric string		Max. 32 characters.
Function	**1 Read Coils** **2 Read Discrete Inputs** **3 Read Holding Registers** **4 Read Inputs Registers** **5 Write Single Coil** **6 Write Single Register** **15 Write Multiple Coils** **16 Write Multiple Registers** **23 Read/Write Multiple Registers**	**3 Read Holding Registers**	When a message is sent from a client to a server device, the function code field tells the server what kind of action to perform.

Parameter	Value	Default	Description
Trigger	**Cyclic** **Data Change** **Disable**		Disable: The command was never sent. Cyclic: The command is sent cyclically at the interval specified in the Poll Interval parameter. Data change: The data area is polled for changes at the time interval defined by Poll Interval. A command is issued when a change in data is detected.
Poll Interval (This will show up when you select trigger mode 'cyclic'.)	100 to 1200000 ms	1000	Polling intervals are in milliseconds. Since the module sends all requests in turns, the actual polling interval also depends on the number of requests in the queue and their parameters. The range is from 100 to 1,200,000 ms.
Endian Swap	**None** **Byte Swap** **Reverse** **Reverse with Byte Swap**	None	**None** (none, follow the protocol's Endianess, here it is Big endian) • The data remains in its original format without changing the order of bytes. • 0x11 22 33 44 55 66 77 88 → 0x11 22 33 44 55 66 77 88 **Byte Swap** • Switch the order of bytes. • 0x11 22 33 44 55 66 77 88 → 0x22 11 44 33 66 55 88 77 **Reverse** • Reverse the order of bytes. • 0x11 22 33 44 55 66 77 88 → 0x88 77 66 55 44 33 22 11 **Reverse with byte Swap** • Reverse the order of bytes first, then switch the order of bytes. • 0x11 22 33 44 55 66 77 88 → 0x77 88 55 66 33 44 11 22
Read Starting Address	0 to 65535	0	Modbus register address.
Read Quantity	Read Coils: 1 to 2000 Read Discrete Inputs: 1 to 2000 Read Inputs Registers: 1 to 125 Read Holding Registers: 1 to 125 Read/Write Multiple Registers: 1 to 125	10	Specifying how many items to read.
Write Starting Address	0 to 65535	0	Modbus register address.
Write Quantity	Write Multiple Coils: 1 to 1968 Write Multiple Registers: 1 to 123 Read/Write Multiple Registers: 1 to 123	1	Specifying how many items to write into.
Fault Protection	Proceed - Keep latest data Proceed - Clear all data bits to 0 Proceed - Set to User-defined value		If the MGate's connection to the other side (server) fails, the gateway cannot receive data, but the gateway will continuously send output data to the Modbus server device. To avoid problems in this case, the MGate can be configured to react in one of the following three ways: Keep the latest data, clear data to zero, set the data bits to user-defined values.

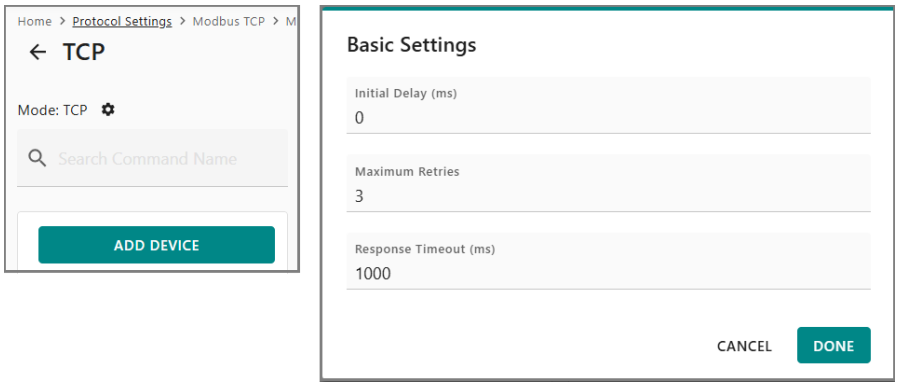
Parameter	Value	Default	Description
User-defined Value (This will show up when you select Fault Protection mode as 'Set to user defined value'.)	00 to FF (Hex)	00 00	The user-defined values to write into the data bits when the Set to user defined value option is selected.
Fault Timeout (This will show up when you select Fault Protection mode as 'Set to user defined value'.)	1 to 86400 ms	3600	Defines the communication timeout for the opposite side.
Tag Type	**raw, boolean, int16, int32, int64, uint16, uint32, uint64, float, double, string**	raw	Specifies the tag data type. The default is raw for fast multiple data mapping. For other data types, you could also scale the resource data. There are two types: • Slope-intercept: tag value = (source value * slope) + offset • Point-slope: tag value = target min + (source value - source min.) * ($\frac{\text{target max. - target min.}}{\text{source max. - source min.}}$)

Modbus TCP Client

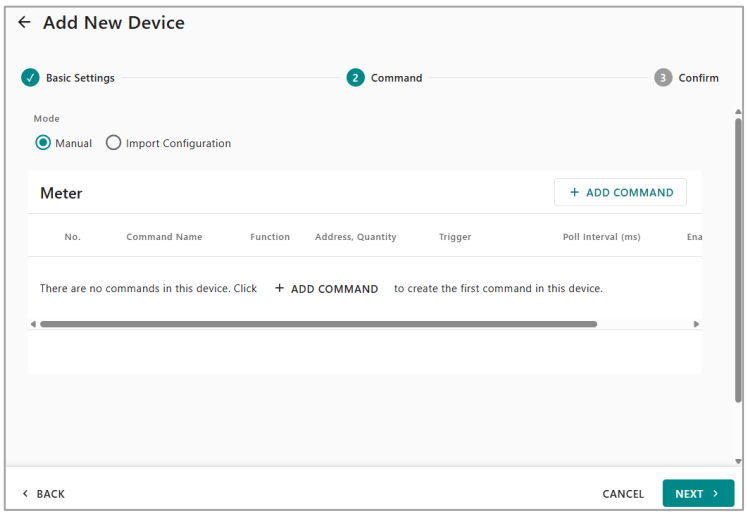
By selecting Modbus TCP in the topology, and selecting on the MGate Modbus TCP client role, you can configure the Modbus commands that MGate will actively poll. The MGate supports csv file import/export for the full Modbus client settings, including command settings; it is easy to use when you back up the settings or offline configuration during the installation stage. We suggest exporting the file first and then importing it back to the MGate after configuring it.



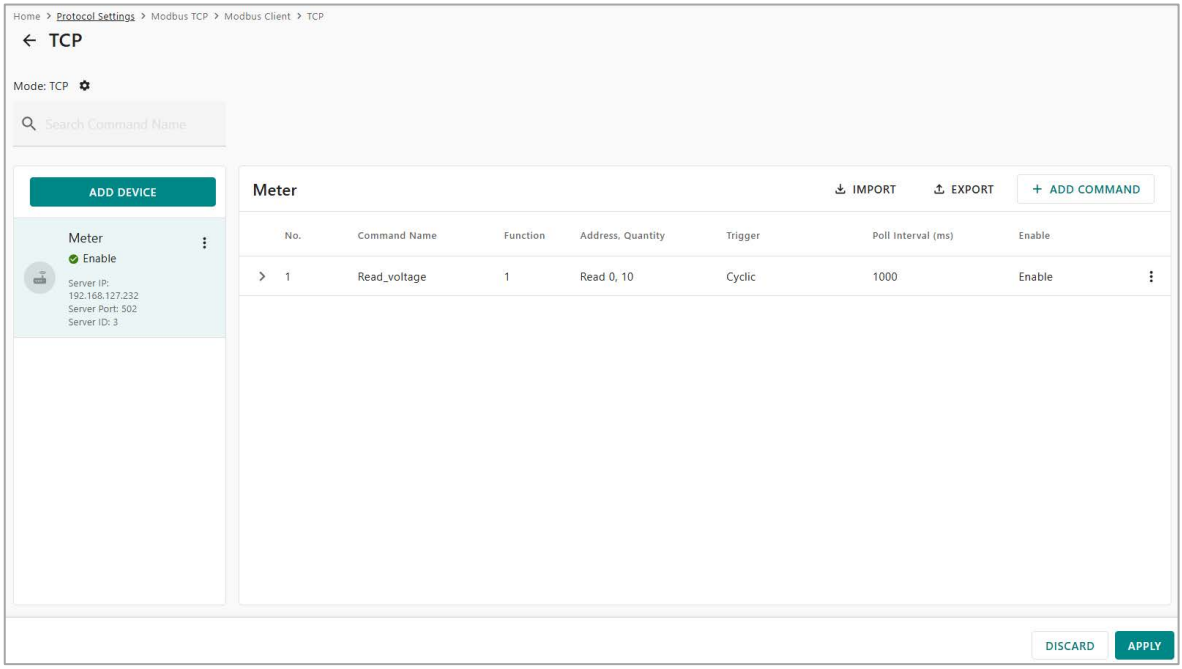
Select TCP, select the **MANAGE** button and then Edit. If you need to change Modbus TCP protocol settings, select the settings besides TCP. Then, you can change the retry or response timeout settings.


Home > Protocol Settings > Modbus TCP > M
TCP
Mode: TCP
Search Command Name
ADD DEVICE
Basic Settings
Initial Delay (ms)
0
Maximum Retries
3
Response Timeout (ms)
1000
CANCEL
DONE

Add a Modbus device by selecting **ADD DEVICE**, set up the name and the target Modbus server name, IP address, Modbus ID, and server port that you are going to ask, and select **ADD COMMAND** to add the Modbus commands. The MGate supports csv file import/export for Modbus command settings. We suggest exporting the file first and then importing it back to the MGate after configuring it.



Confirm your settings and select **DONE**. Finally, remember to select **APPLY** to take effect.



The Modbus command parameters are the same as Modbus RTU/ASCII client. Refer to the table above under the Modbus RTU/ASCII Client section.

Serial Proprietary Client

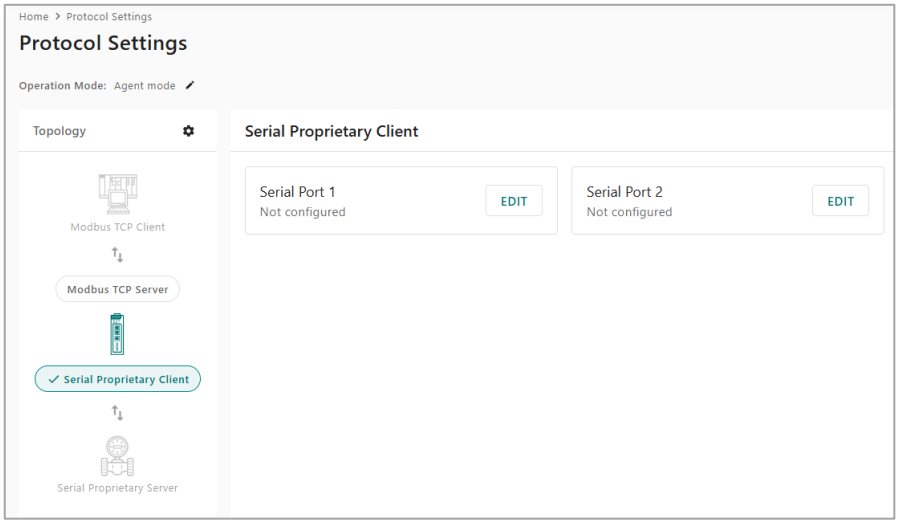
By selecting Serial Proprietary in the topology, and selecting the MGate Serial Proprietary client role, you can configure the serial commands that MGate will actively poll.

You can use different function blocks to create **request/response** or **produce/consume** payloads that comply with the serial device specifications.

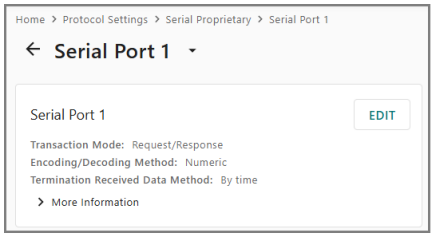
The MGate supports the following types of function blocks to create your serial proprietary commands:

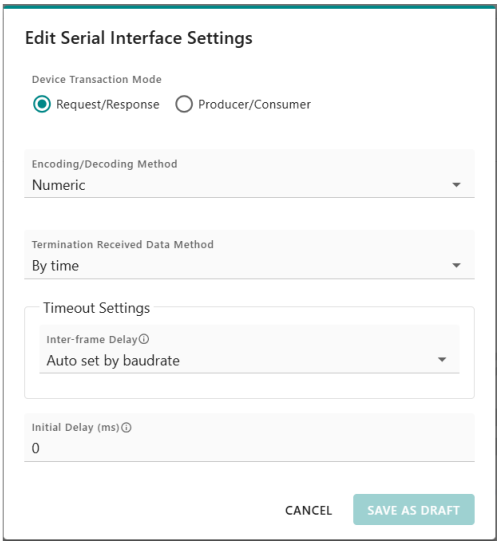
- **Node Address:** Used when the device scenario requires a Node Address (Modbus ID).
- **Constant:** Used when the device scenario requires a fixed pattern that is not intended for data exchange.
- **Data:** Used when data needs to be exchanged between northbound and southbound devices.
- **Checksum:** Used when frame checking is required in the device scenario.

To configure commands for serial proprietary, select the desired serial port and **EDIT**.



To change the serial port's general settings, such as transaction mode, encoding, select **EDIT**. The parameters are explained below.

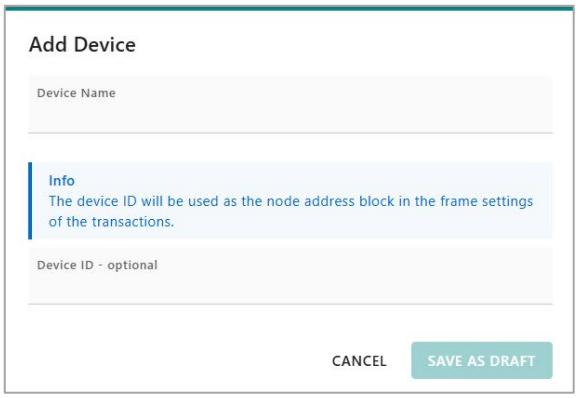




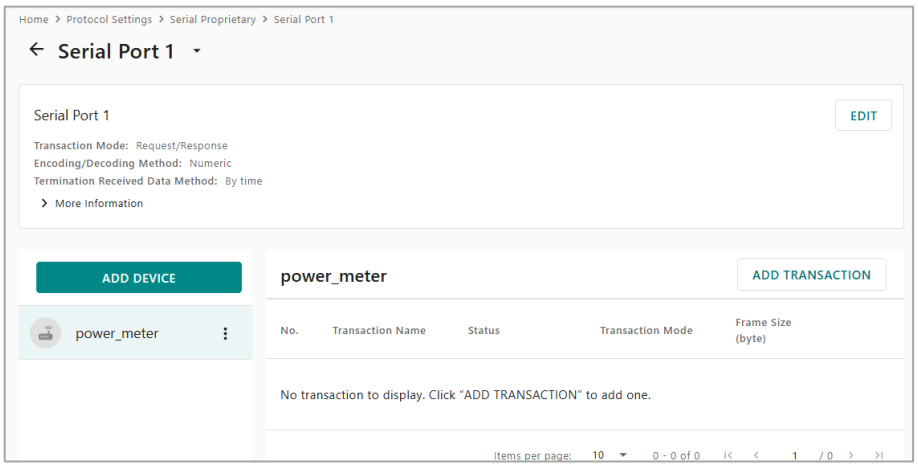
Parameter	Value	Default	Description
Transaction Mode	Request/Response; Producer/Consumer	Request/ Response	Define the communication mode for devices under this interface as either Request/Response or Produce/Consume. • **Request/Response:** One device **asks** (sends a request), and the other device **answers** (sends a response). Example: Like a customer ordering food at a restaurant — you ask, they reply with your meal. *(Common in Modbus-like, proprietary Modbus, Modbus extensions.)* • **Producer/Consumer:** One device **keeps sending data** (producing), and other devices **just receive it** (consuming), without being asked each time. Example: Like a train station broadcasting information — it keeps sending, and any device can listen. *(Common in real-time or event-driven systems.)*
Encoding/ Decoding Method	Numeric, ASCII, Modbus ASCII	Numeric	Used to determine whether the encoding/decoding method within the frame for transmission and reception on this device is **Numeric, ASCII** or **Modbus ASCII**. **Example:** The number 1234 can be represented in actual data transmission as: • **Numeric:** 0x04D2 • **ASCII:** 0x31 0x32 0x33 0x34 Generally, protocols similar to Modbus RTU use **Numeric** encoding, while some serial ASCII protocols use **ASCII** encoding. Example of a serial ASCII protocol data: A@=@ **Modbus ASCII:** means it is a proprietary protocol based on Modbus ASCII The differences and length of these three methods are below: **Device ID = 255** • **Numeric:** Frontend sends 255 to the backend, **length = 1** (actual transmission: 0xFF) • **ASCII:** Frontend sends 255 to the backend, **length = 3** (actual transmission: 0x32 0x35 0x35) • **Modbus ASCII:** Frontend sends 255 to the backend, **length = 2** (actual transmission: 0x46 0x46)
Termination Received Data Method	**By time, By delimiter**	By time	Defines the method used to segment a complete frame when receiving packets. For example, Modbus RTU-like protocols use time to separate frames.
Inter-frame delay	**(If Termination Received Data Method is set to by time) Auto-set by baudrate, User-defined value**	**Auto-set by baudrate**	Default: auto set by baudrate, can be user-fined between 10 to 500 ms
Delimiter Type	**(If Termination Received Data Method is set to by delimiter) Start delimiter, End delimiter, Start delimiter and end delimiter**	**Start delimiter**	**Start delimiter:** When the specified start character is received, it is identified as the beginning of a new frame. Up to two bytes can be configured, with the second byte being optional. **End delimiter:** When the specified end character is received, it is identified as the end of a new frame. Up to two bytes can be configured, with the second byte being optional. **Start delimiter and end delimiter:** You can configure both of the above. Enter Hex for one or two bytes of the delimiter

Parameter	Value	Default	Description
Initial delay	**0~120s**	0	Some serial devices may take more time to boot up than other devices. In some environments, this may cause the entire system to experience repeated exceptions during the initial boot-up. After booting up, you can force the MGate to wait before sending the first request with the Initial Delay setting.

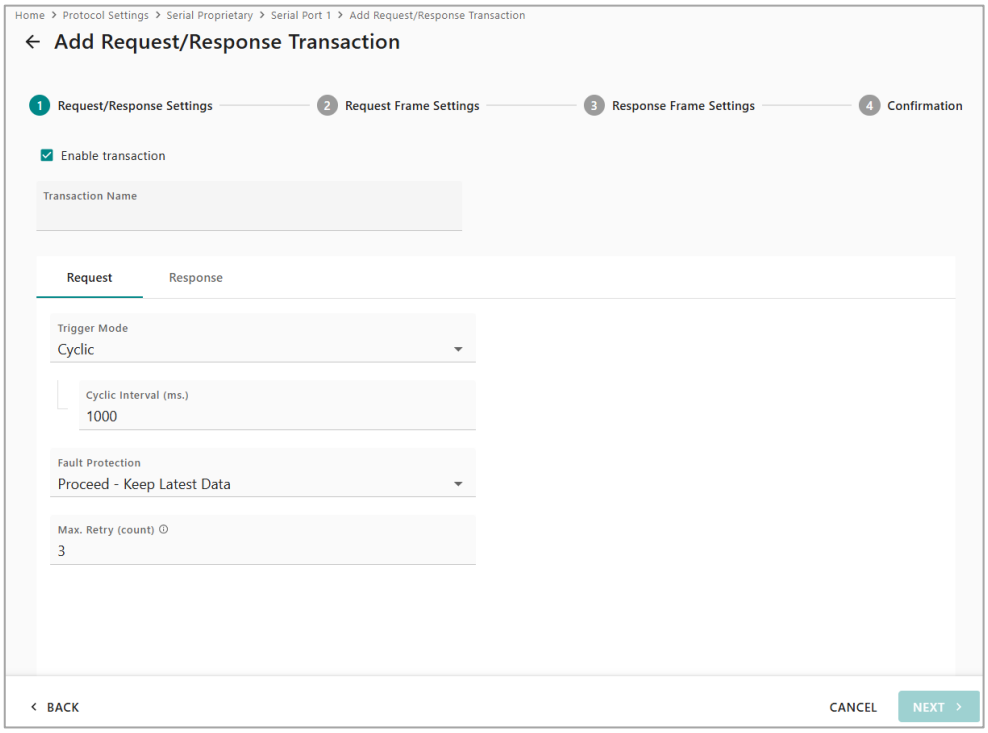
Add a serial device by selecting **ADD DEVICE**, set up the name and the device ID. Device ID is optional if your protocol does not need this field. Select **SAVE AS DRAFT** to save this device template.



After adding a device, select **ADD TRANSACTION** to create communication commands or transmission payloads.



When creating transactions, go from **Request/Response settings > configure Request frame > configure Response frame** to finish it.

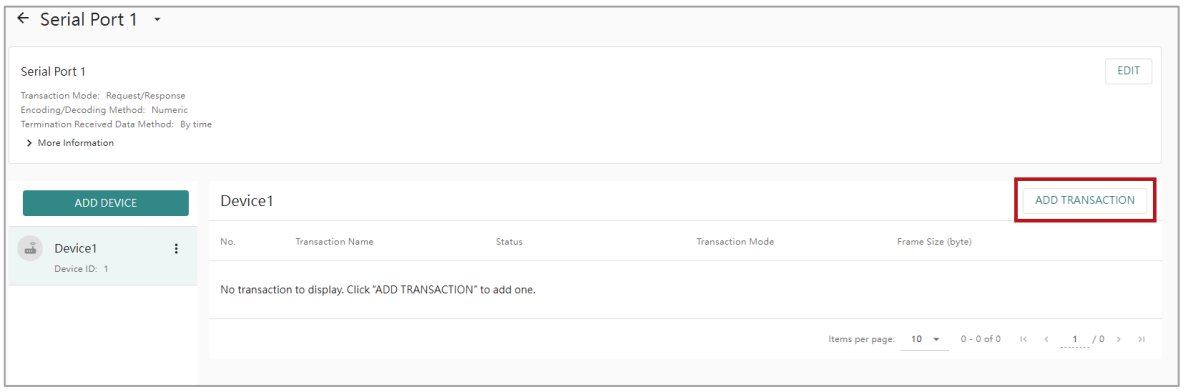


For **Request/Response settings**, give a Transaction Name to identify the function of this command or operation. The following are parameters that can be set in Request settings.

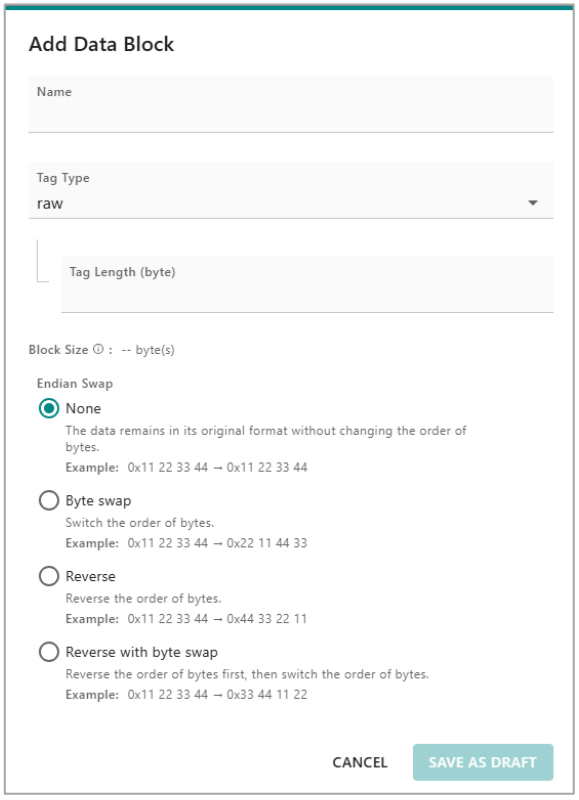
Parameter	Value	Default	Description
Trigger	**Cyclic** **Data Change**		Cyclic: The command is sent cyclically at the interval specified in the Poll Interval parameter. Data change: The data area is polled for changes at the time interval defined by Poll Interval. A command is issued when a change in data is detected.
Cyclic Interval (This will show up when you select trigger mode 'cyclic'.)	100 to 1200000 ms	1000	Polling intervals are in milliseconds. Since the module sends all requests in turns, the actual polling interval also depends on the number of requests in the queue and their parameters. The range is from 100 to 1,200,000 ms.
Fault Protection	Proceed - Keep latest data Proceed - Clear all data bits to 0 Proceed - Set to User-defined value		If the MGate's connection to the other side (server) fails, the gateway cannot receive data, but the gateway will continuously send output data to the Modbus server device. To avoid problems in this case, the MGate can be configured to react in one of the following three ways: Keep the latest data, clear data to zero, set the data bits to user-defined values.
Max. Retry (count)	**0~5**	3	The MGate will resend this transaction when a response timeout occurs.

For Response, you can configure the response timeout. Default is 1000 ms.

To **configure Request frame**, select the **ADD FUNCTION BLOCK** to add a Node address block, Constant block, a Data block, or a Checksum block. See the parameters for these function blocks. If you have configured a device ID previously, then you will see a Node address block available here.



Data Block



Parameter	Value	Default	Description
Name	Alphabetical string up to 64 characters		A name to identify this data block
Tag Type	Raw, Boolean, int8, int16, int32, int64, uint8, uint16, uint32, uint64, float, double	**raw**	Provides multiple formats to accommodate the data types of serial proprietary devices.
Tag Length (only configurable when tag type is raw)	1-500 (Byte)		The length of this tag. Note: the Boolean type tag uses 1 byte.

Parameter	Value	Default	Description
Endian Swap	None, Byte swap, Reverse, Reverse with byte swap	None	**None:** The data remains in its original format without changing the order of bytes. • Example:0x11 22 33 44 → 0x11 22 33 44 **Byte swap:** Switch the order of bytes. • Example:0x11 22 33 44 → 0x22 11 44 33 **Reverse:** Reverse the order of bytes. • Example:0x11 22 33 44 → 0x44 33 22 11 **Reverse with byte swap:** Reverse the order of bytes first, then switch the order of bytes. • Example: 0x11 22 33 44 → 0x33 44 11 22

Constant Block

For the Constant block, when you are using **Numeric/Modbus ASCII** encoding, you can only enter **HEX**.

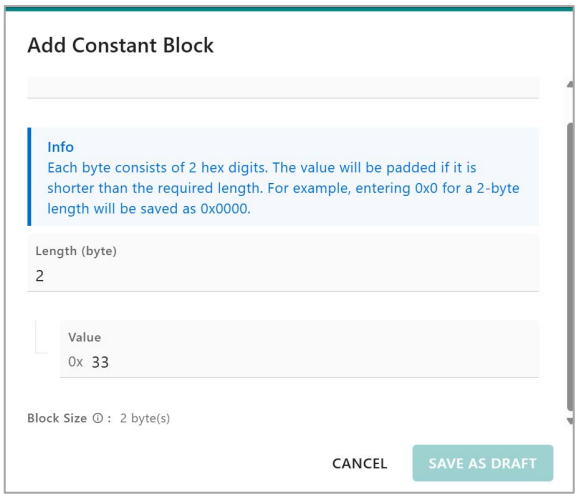
Example: User inputs 0x00FF

- **Numeric:**
  When you enter 00FF, the data **length = 2** (the MGate will send: 0x00FF)
- **Modbus ASCII:**
  When you enter 00FF, the data **length = 4** (the MGate will send: 0x30 0x30 0x46 0x46)

When using **ASCII** encoding, you can only enter **ASCII** characters or select from a **dropdown escape character menu** (Hex values cannot be entered directly).

Example: User inputs @123

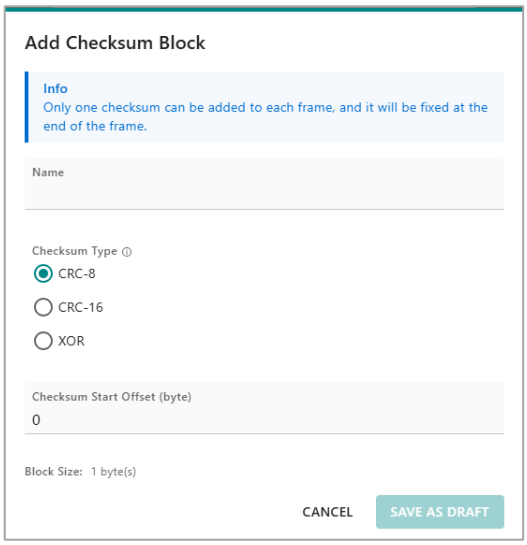
The MGate will send 40313233 and the data **length = 4**.



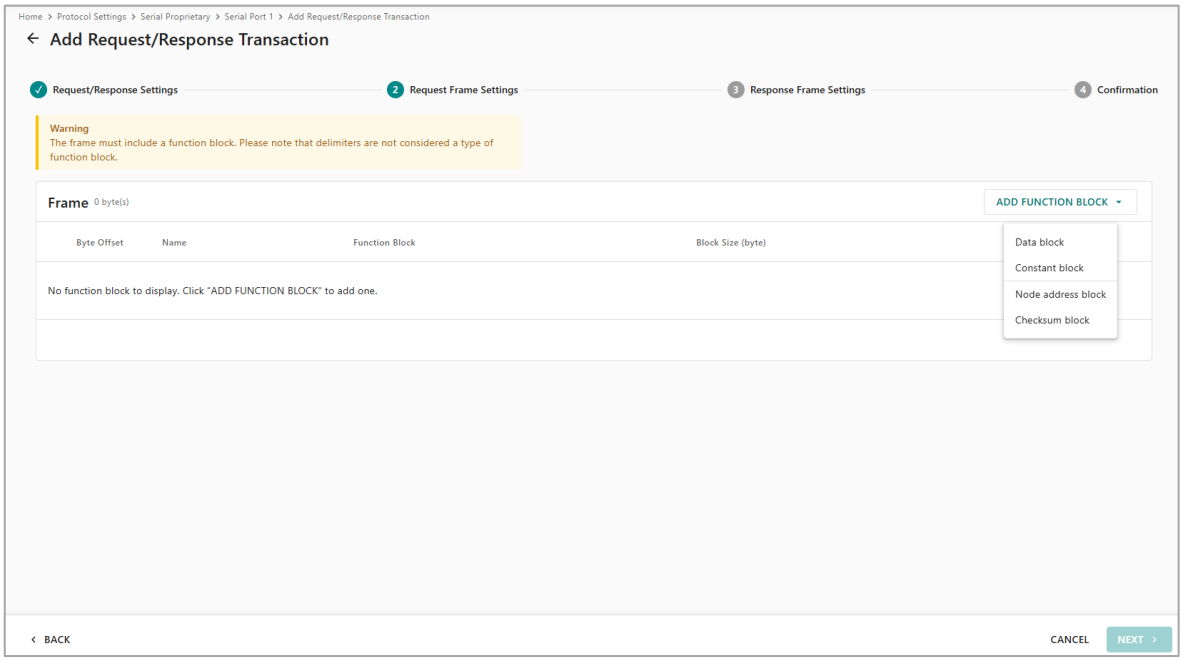
Parameter	Value	Default	Description
Name	Alphabetical string up to 32 characters		A name to identify this constant block
Length	1-500 (byte)		The length of this constant. Each byte consists of 2 hex digits.
Value	Hex value		The value of this constant. The value will be padded if it is shorter than the required length. For example, entering 0x0 for a 2-byte length will be saved as 0x0000.

Checksum Block

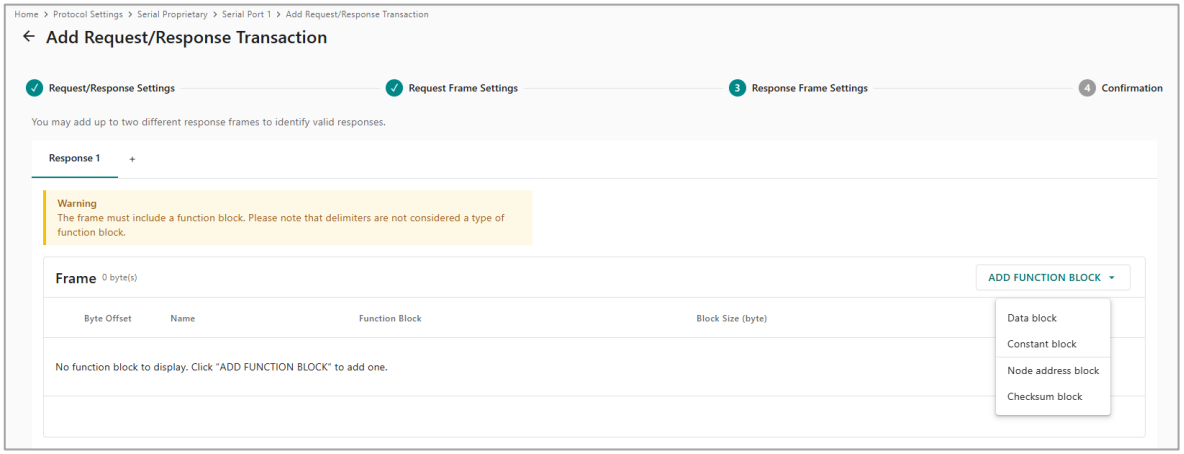
Only one checksum can be added per frame, and it must be placed as the last function block (before the delimiter).

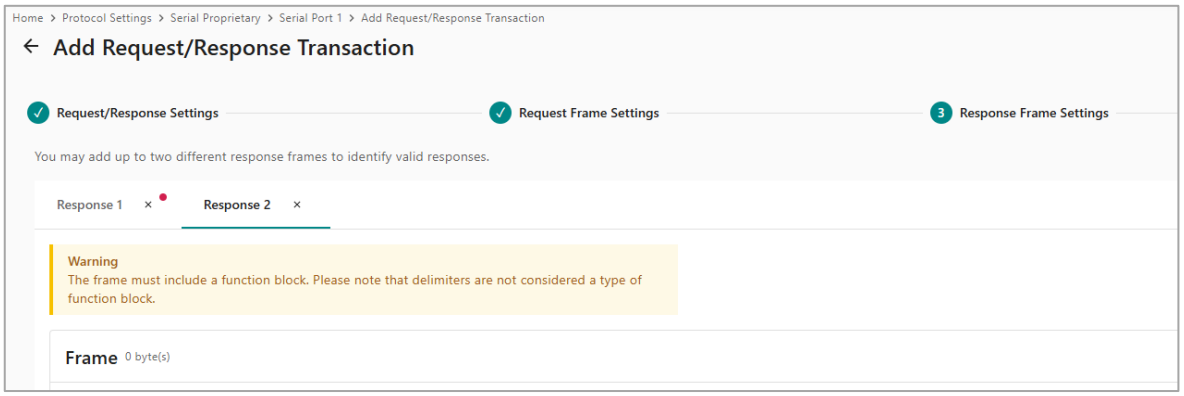


Parameter	Value	Default	Description
Name	Alphabetical string up to 32 characters		A name to identify this checksum block
Checksum Type	CRC-8, CRC-16, XOR LRC is only enabled when using Modbus ASCII encoding	CRC-8	Provides a checksum mechanism to ensure data integrity, with the calculation based on all data except the checksum itself.
Checksum start offset (byte)	**Numerical number**	0	**Specifies from which byte the checksum calculation begins.** **Example:** 0 means the calculation starts from the first byte up to the byte before the checksum. The start delimiter is also considered part of the frame, so an offset of 0 means the delimiter is included in the calculation.



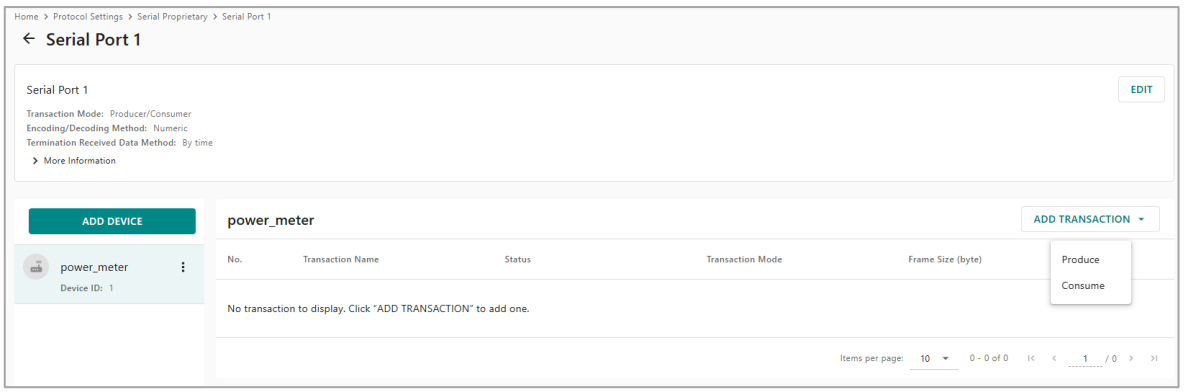
To **configure Response frame**, you can also add a Node address block, Constant block, a Data block, or a Checksum block. You can configure up to two response frames.





When finishing all settings, remember to select **APPLY** to take effect.

If you configure **Producer/Consumer** as the Transaction Mode, you can select **ADD TRANSACTION**, and then you will be able to add Produce or Consume commands or transactions. Same as above, you can also add a Node address block, Constant block, a Data block, or a Checksum block.



The following is an example of configuring request/response transactions for polling an Enron Modbus device.

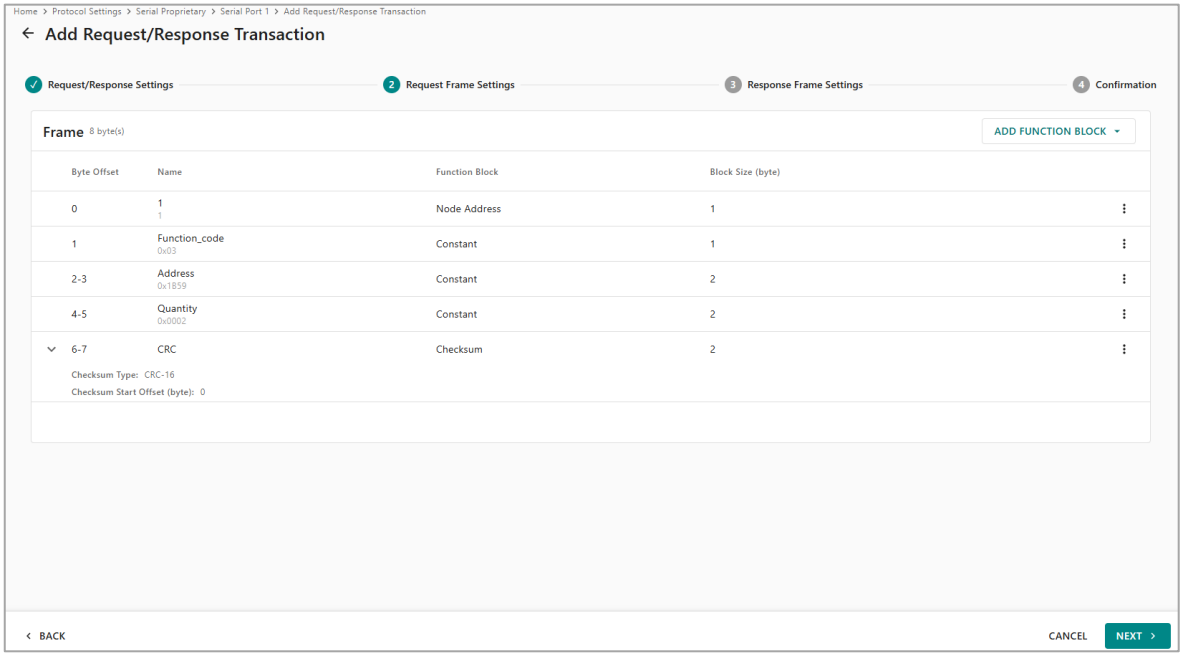
Request Transaction Example

Look at the following instructions:

Data Addresses (hexadecimal)	Register Numbers (decimal)	Type	Table Name
03E9 to **07CF**	1001-1999	Read-Write	Boolean variables
0BB9 to **0F9F**	3001-3999	Read-Write	16 bit Short integer variables
1389 to **176F**	5001-5999	Read-Write	32 bit Long integer variables
1B59 to **1F3F**	7001-7999	Read-Write	32 bit Floating point variables

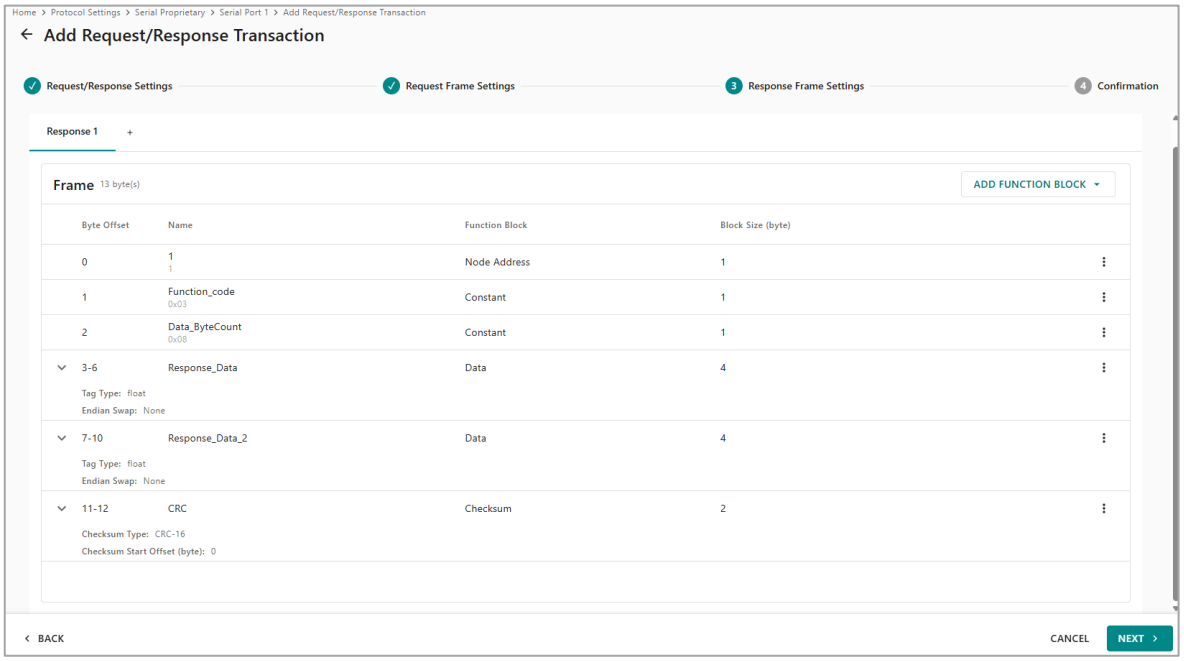
We are reading the register 7001 (float point values). The address will be filled as 0x1B59.

As we are reading float, the response data tag type also needs to be set to float.



Response Transaction Example

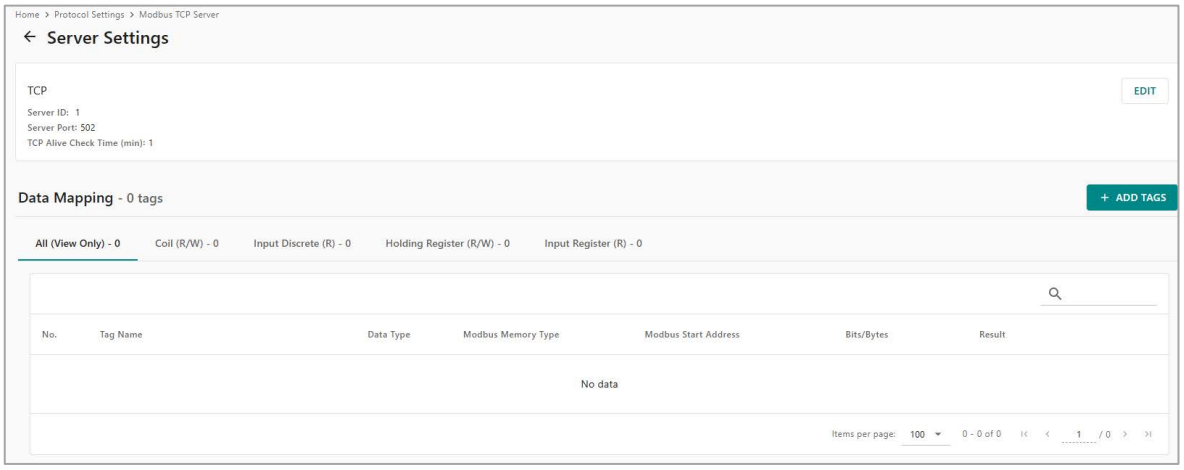
This is an example for Enron Modbus response.



Modbus TCP Server

By selecting Modbus TCP in the topology, and selecting on the MGate Modbus TCP server role, you can configure the Modbus TCP listen settings. The parameters are below.

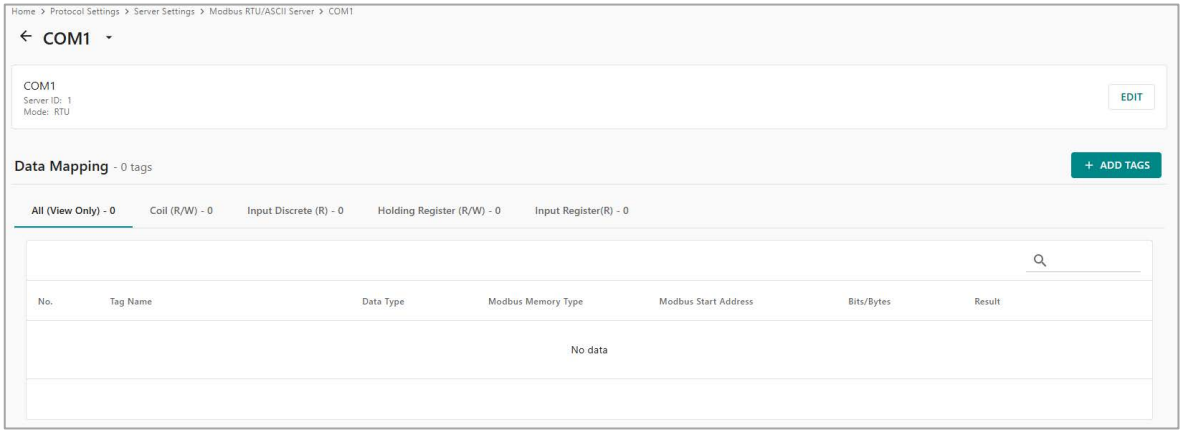
Parameter	Value	Default	Description
Server ID	1 to 255		The Modbus server ID.
Server Port	1 to 65535	502	The Modbus TCP listen port number.
TCP Alive Check Time (min)	0 to 99	1	The MGate will check when it received the last Modbus TCP packet. If it exceeds the timeout (default 1 min), it will reset the TCP session to avoid the TCP session being occupied.



To finish mapping, select **ADD TAGS** from the other side (Edge side) protocol.

Modbus RTU/ASCII Server

By selecting Modbus RTU/ASCII in the topology and selecting on the MGate Modbus RTU/ASCII server role, you can configure the Modbus RTU/ASCII settings. The parameters that can be configured are server ID and Mode (Modbus RTU or ASCII protocol).



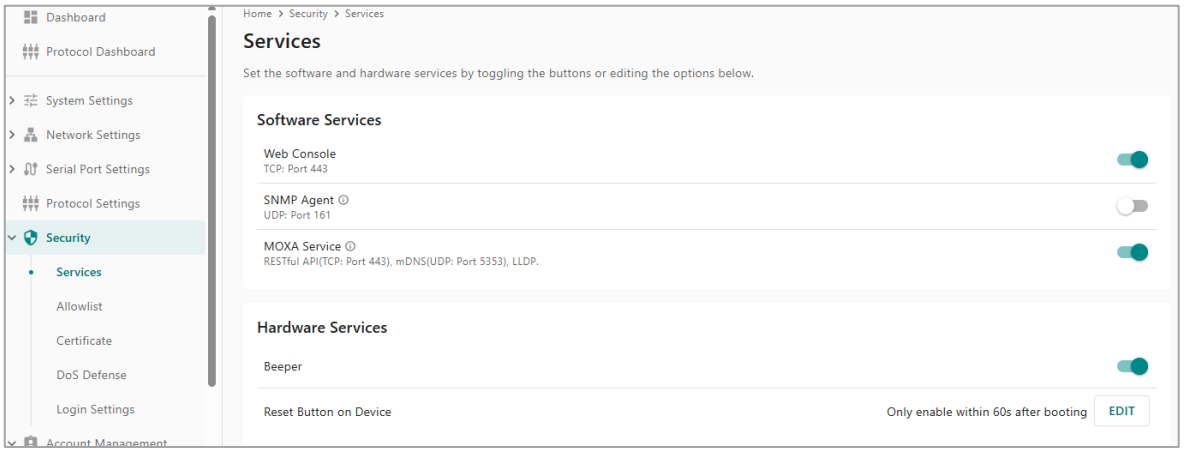
To finish mapping, select **ADD TAGS** from the other side (Edge side) protocol.

Security

With cyberattacks growing in number and sophistication, device server vendors are adding functions geared towards protecting sensitive business and personal information. All the relevant functions are listed under the **Security** category.

Services

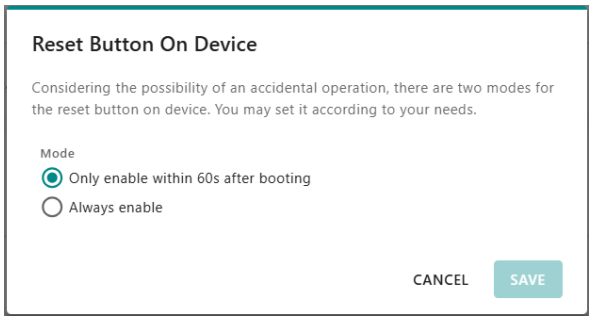
Based on different user scenarios, you may need different services to meet these requirements. Select **Security > Services** to enable/disable the services he needs or no need.



Software Services	Value	Default Value	Description
Web Console	Enable/ Disable	Enable	This setting is to enable/disable the web console. To ensure security, the MGate MB3000-G2 device server only supports HTTPS console using TLS v1.2 or newer. The web console provides all the settings that the MGate MB3000-G2 supports. We don't recommend a user to disable it.
SNMP Agent	Enable/ Disable	Disable	This setting is to enable/disable the SNMP Agent service. If you want to use the SNMP protocol to monitor the status or change some configuration settings of the MGate MB3000-G2, enable the service. If your site doesn't match this scenario, please disable it.
Moxa Service	Enable/ Disable	Enable	This setting is to enable/disable Moxa proprietary service. Windows Driver Manager for Real COM, DSU v3.0, MXview One and MXconfig are based on this service to work. This software cannot be used when Moxa Service is disabled.

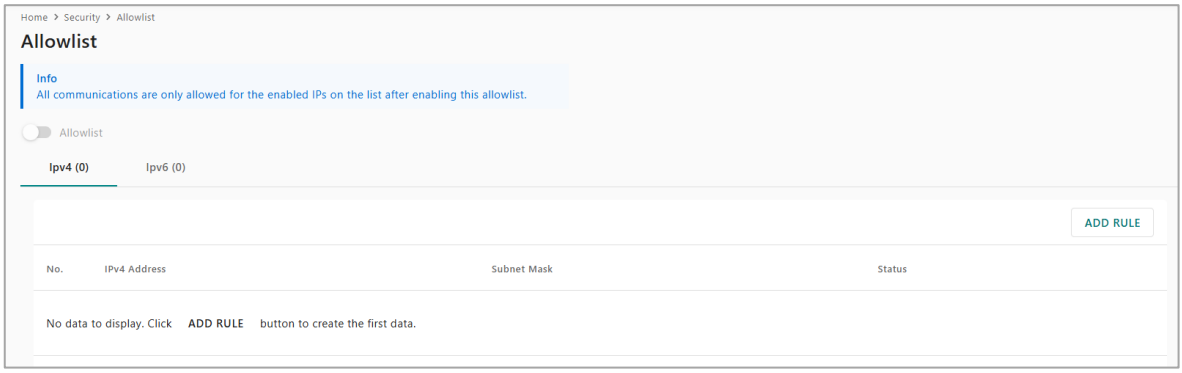
Hardware Services	Value	Default Value	Description
Beeper	Enable/Disable	Enable	This setting is to enable/disable the beeper of the device. You will hear the beeper when the device is ready after a power cycle. If you don't want to hear the sound, you may disable the service.
Reset Button on Device	Only enable within 60s after booting up/Always enable	Only enable within 60s after booting up	By default, the device disables the reset button after booting up for 60 seconds to prevent someone from accidentally pushing the button and resetting the device to its default settings.

The EDIT button in the Reset Button On Device service allows you to specify when the reset button should be enabled. Either the button is enabled for just one minute after the device boots up, or it stays enabled indefinitely.

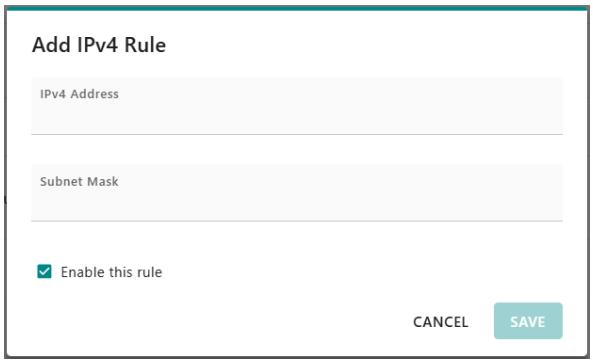


Allowlist

An allowlist is a list of IP addresses or domains that are provided privileged access. Enabling this function limits the number of IP addresses that can access the device server, which can prevent unauthorized access from an untrusted network.



Before you enable the allowlist, add at least one rule to the table. And remember to make sure the host PC's IP address is on the list, or you may not access the web console of the device server.

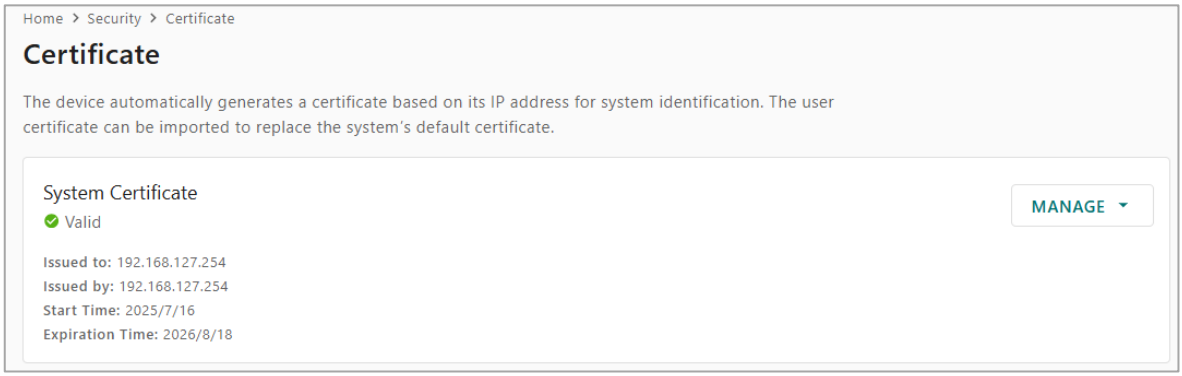


Select the ADD RULE button to add a new rule. You may fill an IP address or a domain name in the IP Address column and then input the subnet mask to allocate a range of IP addresses. We recommend you enable this function so the new rules will be enabled while adding a new rule. If you don't want to enable it, remember to uncheck the checkbox **Enable this rule**.

Certificate

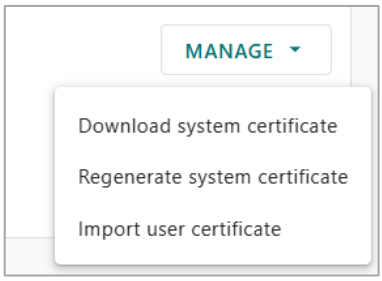
The MGate MB3000-G2 will automatically generate self-certifications for all the TLS sessions, including web console (HTTPS), secure operation modes, and syslog-ng service.

If you have a company-generated or a third-party verified certification, select the **MANAGE** button to import the certification to mitigate the cybersecurity risks to the network.



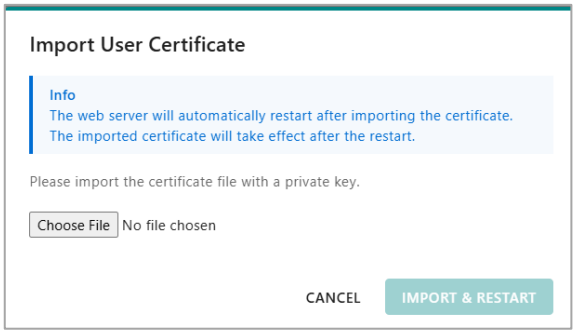
When accessing the **Security > Certificate** page, it shows the status of the system certificate:

- Is the system certificate still valid? Or has it expired?
- Who requested the system certificate?
- Who issued the system certificate? If it is a self-certification, the IP address will be MGate's IP address.
- When was the system certificate issued?
- When will the system certificate expire?



When you select the **MANAGE** button, there are three actions:

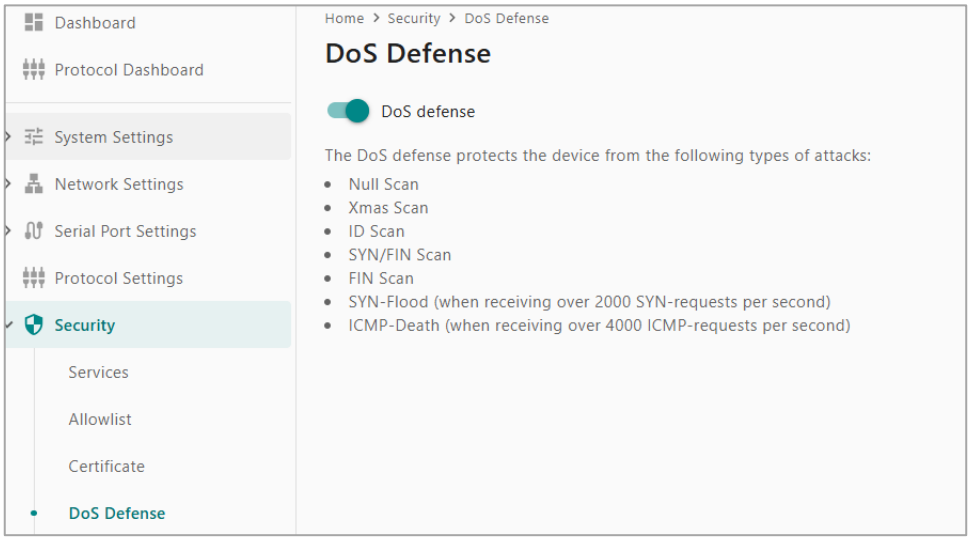
- Download system certificate: The browser or the software on a PC may request the target device to provide a valid certificate before establishing a secure connection. Here, download the system certificate from the MGate. and then upload it to the browser or the software. A secure connection will be established.
- Regenerate system certificate: If the system certificate has expired or is no longer secure, regenerate the system certificate for new secure connections.
- Import user certificate: If you have a company-generated or a third-party verified certification, import that certificate to the MGate to establish new secure connections.



When selecting the **MANAGE > Import user certificate**, select the **Choose File** button to find the certification on the PC. Select the **IMPORT & RESTART** button to ensure the MGate will restart itself to use the imported certificate.

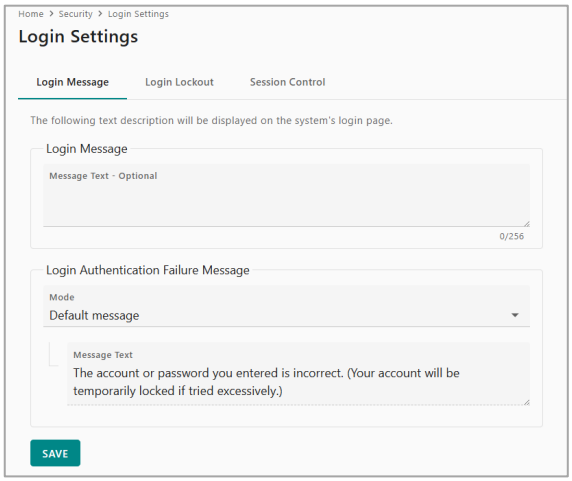
DoS Defense

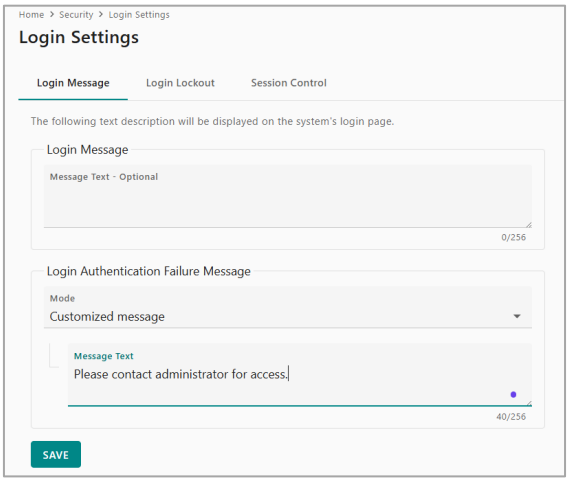
Enable/disable DoS Defense to fend off cybersecurity attacks. A denial-of service (DoS) attack is an attempt to make a machine or a network resource unavailable. The following DoS defense options are enabled by default.



Login Settings

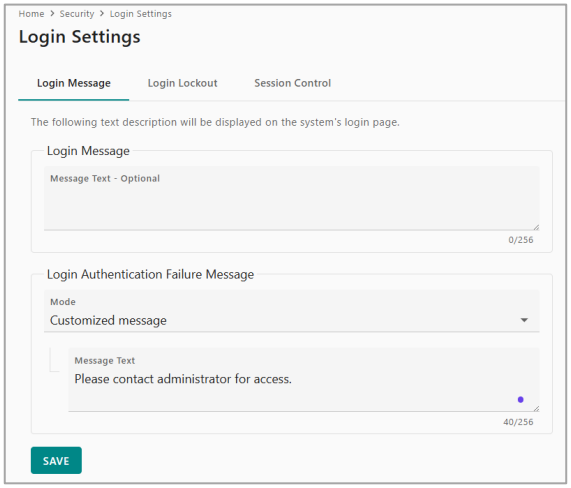
The MGate's administrator may need to send messages to a user upon successful or failed login attempts. The administrator can edit related messages or functions here.





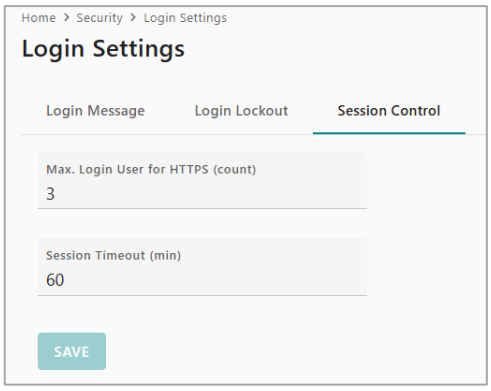
When you successfully log in to an MGate MB3000-G2, the Login Message column will be shown. The message input by the administrator can be up to 256 characters long.

To communicate with users who couldn't log in, the administrator can opt for Customized message mode and enter the message in the Message Text column. When the mode is set to Default message, the MGate also offers a recommended message for the administrator to refer to.



To prevent hackers from repeatedly attempting to log in and crack passwords, we recommend you enable the Login Lockout function. It will be enabled by default.

Name	Value	Default Value	Description
Enable login failure lockout	Checked/ Unchecked	Checked	When checked, the Login Lockout function will be enabled.
Max. Failure Retry (times)	1-10	5	If the Login Lockout function is enabled, it sets the number of attempts before a user is locked out. Let's say the value is 5, then, five password attempts are allowed. Regardless of whether the password is right or wrong on the sixth attempt, access to the device will be denied.
Enable reset login failure counter	Checked/ uncheck	Unchecked	If this function is enabled, the user can wait a bit and then retry logging in. If this feature is turned off, the only option is to contact the administrator and request an account unlock.
Lockout Time (min)	1–60	5	If the option to reset the login failure counter is turned on, it sets the waiting time for the user before another login attempt.



For security and resource arrangement reasons, the MGate will limit the usage of the HTTPS sessions.

Name	Value	Default Value	Description
Max. Login User for HTTPS (count)	1–10	5	The number of users with different user accounts that can establish a HTTPS connection to the MGate.
Session Timeout (min)	1-1440	60	The time the MGate allows for inactivity when a user logs in before ending the HTTPS session.

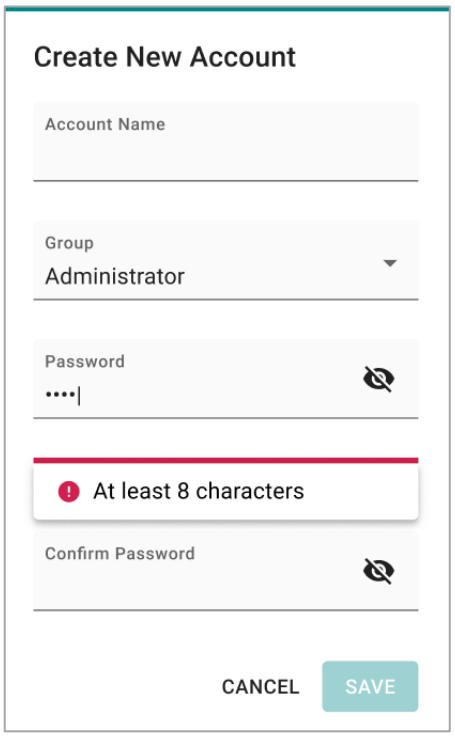
Account Management

For security concerns, different users need different accounts and privileges on one device. With the Account Management function of the MGate MB3000-G2 Series, administrators can easily add, delete, or change user account names. They can also assign access to specific function categories based on different user groups. Furthermore, administrators can effectively manage passwords and login policies to ensure that only authorized users can use the device.

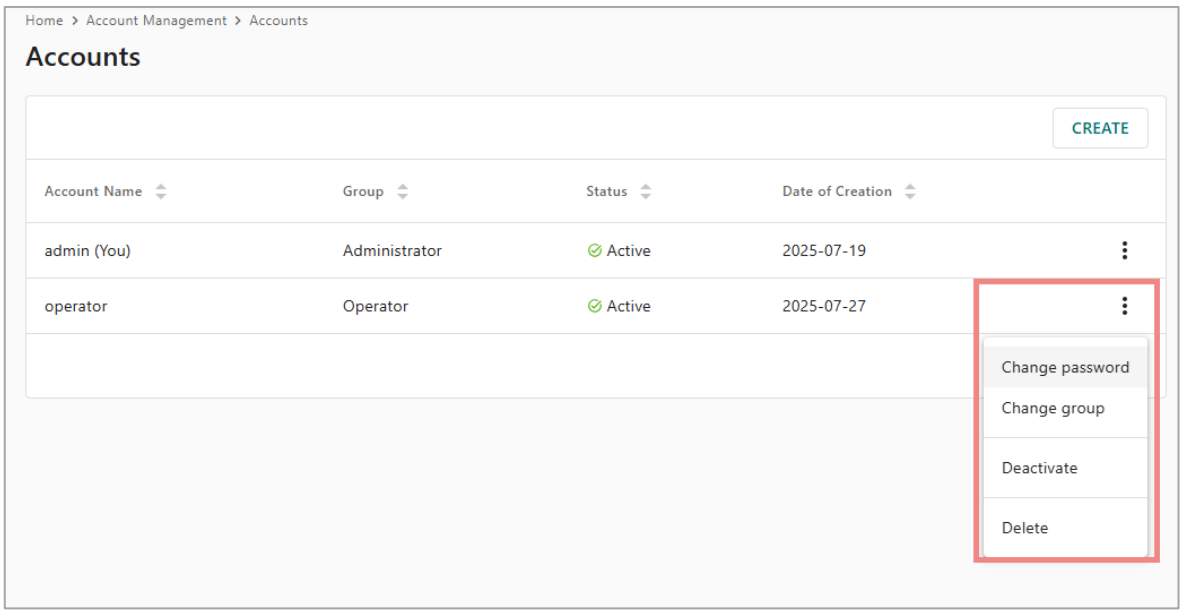
Accounts

In the MGate MB3000-G2 Series, the categories that you can access have a strong correlation with the user groups defined by the administrator(s) (for managing the groups, referring to the next section, Groups). Administrators are allowed to add user accounts to the MGate MB3000-G2 device by selecting the **CREATE** button on the Accounts page.

The **Create New Account** window will pop up for you to input account information and assign a password to the user. Also, the Administrator(s) shall assign a proper **Group** to users to limit their privileges of using the MGate MB3000-G2. To add/delete/edit the **Group**, go to the **Groups** section in the menu. The **Password** rules can be set up in **Password Policy** section.

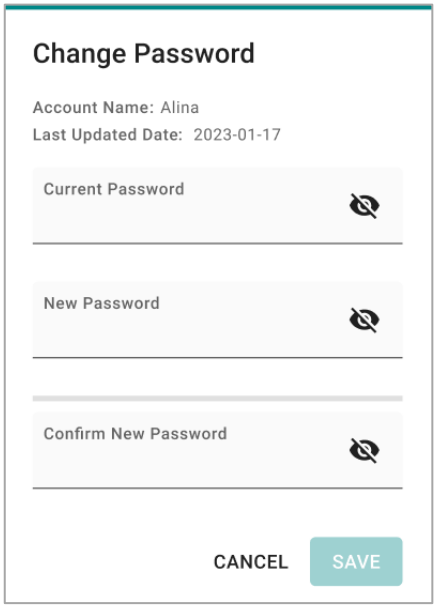


You can also select the **More** menu button on an exited user account to edit the account's password/group, deactivate the account, or delete the account.

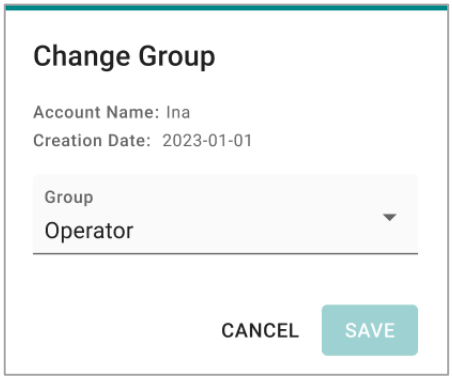


Changing Passwords

As an administrator, you can change every user's password. The Change Password window will appear. Input the new password twice and **SAVE** the new password. The password will be changed. As a general user, you can only change your password. Select the More menu button in your account name and select **Change password** so that the Change password window opens. Input the new password twice and **SAVE** the new password. The password will be changed.

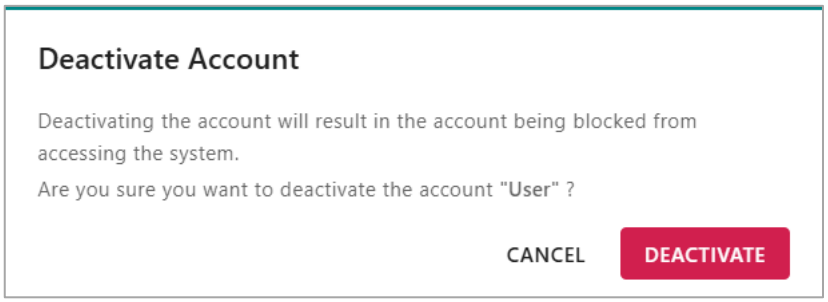


Change Groups



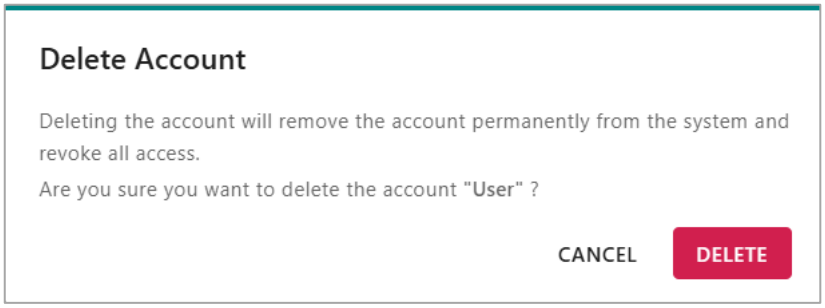
Only the administrator can change the group of a user account. Select the More menu button in the target account name and select Change Group to open the **Change Group** window. On the drop-down menu, select the group you want to move by selecting the **SAVE** button. The user account will move to a new group.

Deactivate



Only the administrator deactivates a user account. When deactivating a user, the user account still exists on the MGate, but the user cannot log in to the device. Only when the administrator activates the user account can the user log in. Select the **More** menu button on the target account name and **Deactivate** to open the Deactivate Account window. Select the **DEACTIVATE** button and the user account will be deactivated.

Delete



Only the administrator can delete a user account. When deleting a user account, it will be removed from the MGate. Select the More menu button on the target account name. Select **Delete** to open the Delete Account window. Select the Delete button to delete the user account.

Groups

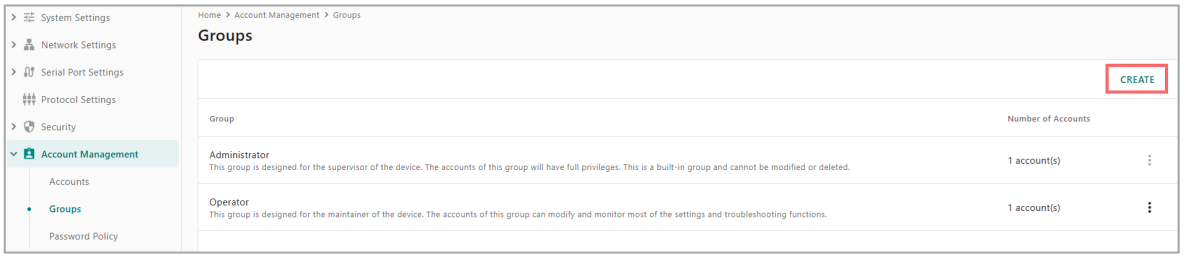
Users can access different function categories with the MGate MB3000-G2 based on their group affiliation.

Customizing access permissions for different groups is restricted to the group administrator by default, or any group that is granted with Read/Write permission on the Account Management category.

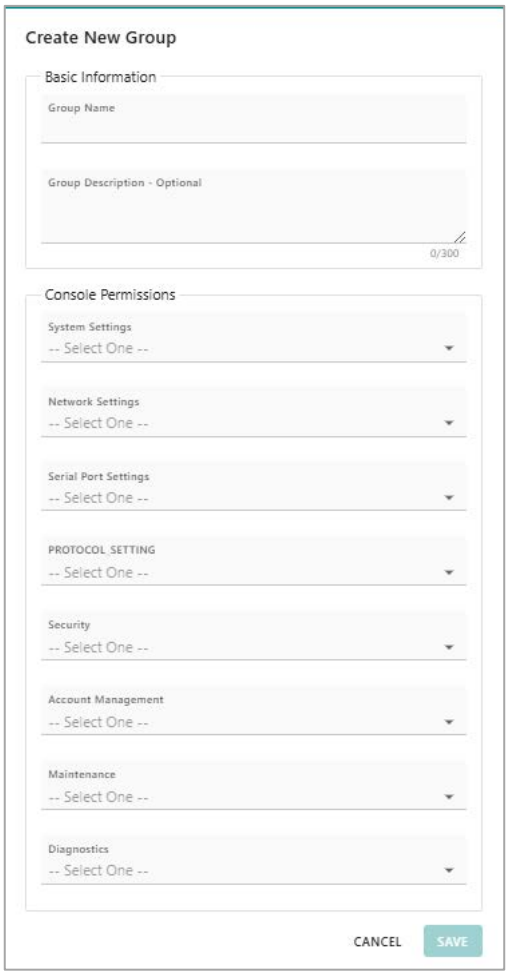
A maximum of four user groups can be created, with up to four user accounts per group. By default, the MGate MB3000-G2 has the Administrator, Operator, and Viewer user groups built in.

- The Administrator group cannot be removed, and the name cannot be changed.
- The Operator group can be removed, and the name can be changed.
- The Viewer group cannot be removed, but the name can be changed.

Select the **Create** button on the Groups page to create a new group.



See below the parameters that can be configured for a group.



Group Name: The name of the group user is going to be created. You should give the group a name.

Group Description—Optional: Describe the group to understand the purpose of creating this group. For example, creating a group named "Operator" with the description: "This group is designed for the maintenance of the device. The accounts of this group can change and monitor most of the settings and troubleshooting functions." This is an optional column.

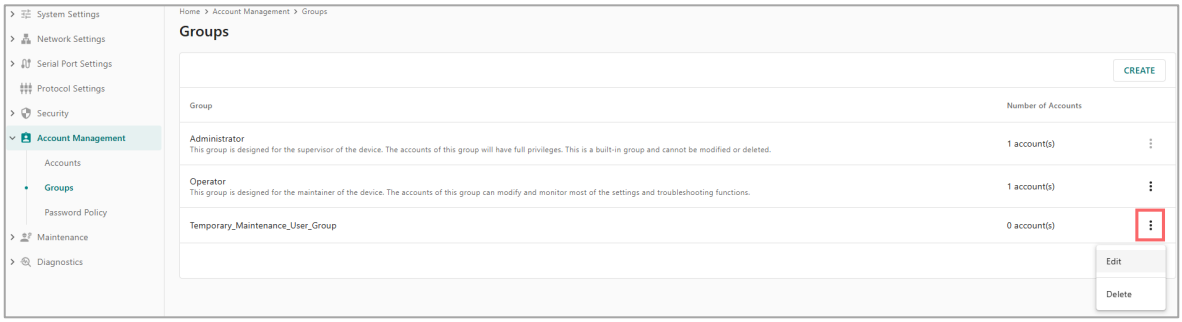
Console Permissions: Assign the privileges for different categories using the drop-down menu. There are three permissions:

- No Display: The user in this user group will not see this function group when accessing the MGate MB3000-G2.
- Read Only: The user in this user group can only view the function/setting in this function group but cannot make modifications.
- Read Write: The user in this user group can view the function/setting in this function group and make modifications.

There are eight functions of the MGate that users can configure the permission:

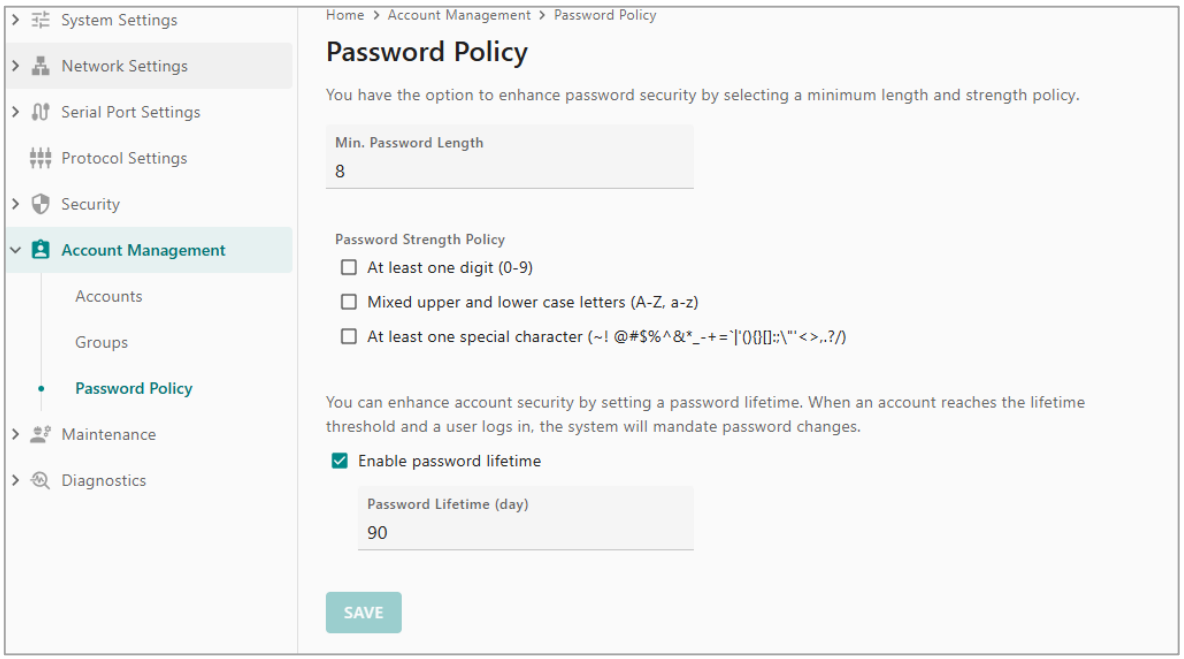
- **System Settings:** Includes all the settings for the MGate, like the server's name and notification.
- **Network Settings:** Includes all the settings related to the Ethernet port, like the IP address and subnet mask.
- **Serial Port Settings:** Includes all the settings related to the serial port, like the operation mode and serial parameters.
- **Protocol Settings:** Includes the settings related to protocol conversions, including Modbus settings.
- **Security:** Includes all the settings related to cybersecurity, like the allowlist and login settings.
- **Account Management:** Includes all the settings related to account and group, like create/modify/delete an account or group.
- **Maintenance:** Includes all the settings related to routine maintenance jobs, like firmware upgrade and configuration import/export.
- **Diagnostics:** Includes all the functions that help the user troubleshoot, like device status and traffic monitoring.

Select the More menu button on an existing group to edit its access privileges or delete the group.



Password Policy

As cybersecurity concerns rise these days, users worry about the risk of password brute-force attacks. The administrator can mitigate cybersecurity risks by enabling the Password Policy function to boost password complexity.



Parameter	Setting	Default	Description
Password minimum length	8 to 256 characters	8	Define the minimum length of the login password for the MGate MB3000-G2
At least one digit (0-9)	Enable/ Disable	Disable	The password must contain at least one number (0 to 9) when enabling this parameter
Mixed upper- and lowercase letters (A-Z, a-z)	Enable/ Disable	Disable	The password must contain an upper- and a lowercase letter when enabling this parameter
At least one special character (~!@#$%^&*-_|;:,.<>[]{}())	Enable/ Disable	Disable	The password must contain at least one special character when enabling this parameter
Enable password lifetime	Enable/ Disable	Enable	Enhancing account security by setting a password lifetime
Password Lifetime (day)	1 to 180 days	90 days	Users can set a specific lifetime for their passwords and receive system notifications to change them if the option is enabled

On completion of the settings, select the **SAVE** button to save the changes and make them effective.

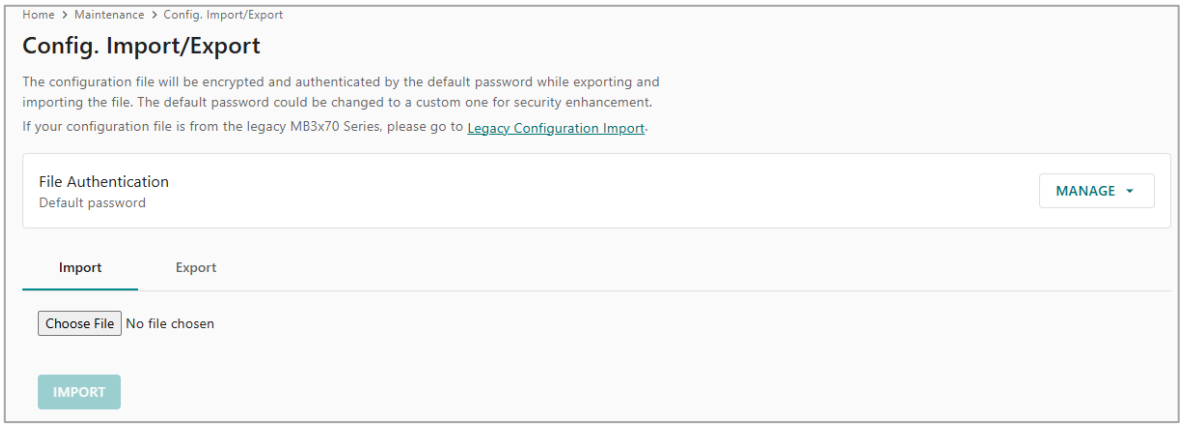
For setting related to failure logins, for example, to lock out an IP address after five failure password inputs, find the **Security > Login Settings > Login Lockout** section.

Maintenance

Operators may have to perform routine tasks every month or quarter to maintain the system when it is online. The MGate categorizes these actions as Maintenance to simplify their completion for you.

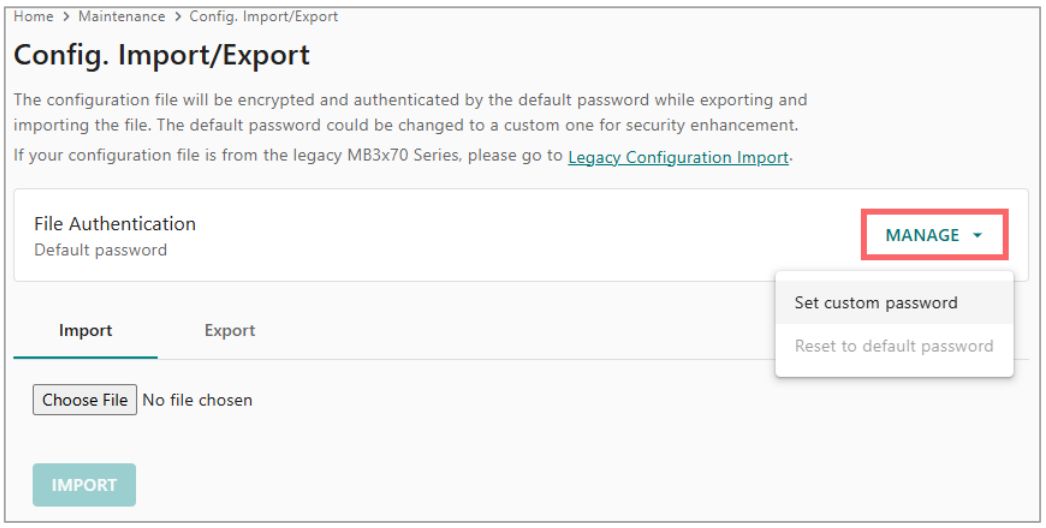
Config. Import/Export

You may want to back up the configuration settings of the MGate to access the **Maintenance > Config. Import/Export** to accomplish it.



Configuration File Authentication

Because of security concerns, the MGate MB3000-G2 provides users with setting up a password to export configuration files. Select the **MANAGE** button to set a password for the exported configuration file.

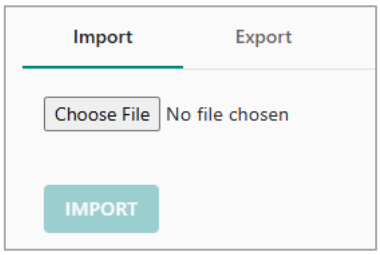


When selecting the **Set custom password**, you can set a customized password for the exported configuration file. The MGate will use this password to decode the imported configuration file. The password policy for the configuration file allows for 8 to 64 characters and does not have any complex requirements.

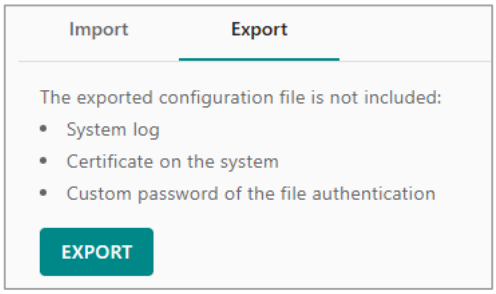
When selecting the **Reset to default password**, the MGate will use the default password to encode or decode a configuration file.

Import/Export the Configuration File

At the **Import** tab, select the **Choose File** button for the configuration file you want to import.



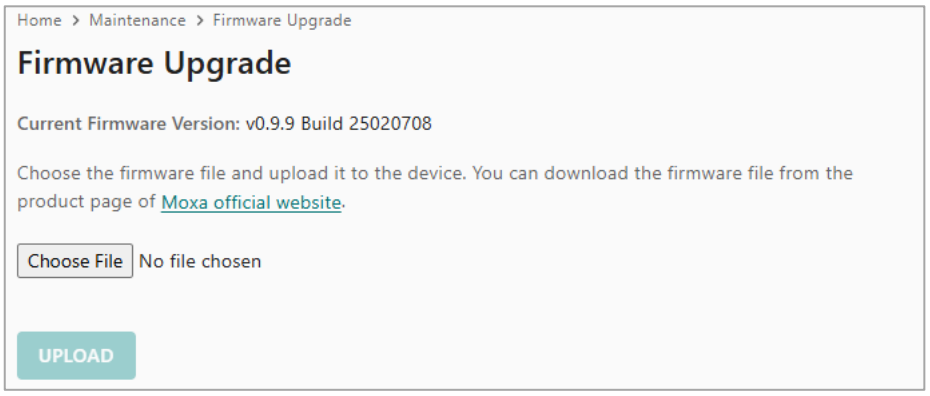
At the **Export** tab, select the **EXPORT** button to determine where you want to save the configuration file to.



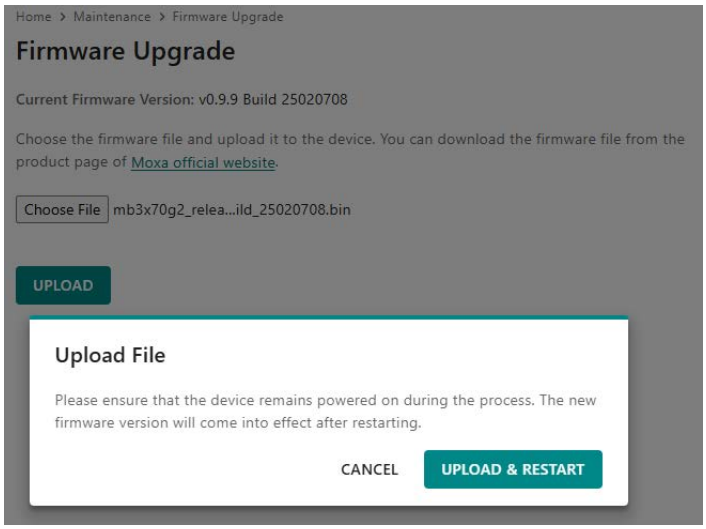
Firmware Upgrade

It's highly advised to always upgrade to the latest firmware version because of the increasing number of cybersecurity threats. Consistently using the latest firmware helps reduce cybersecurity risks.

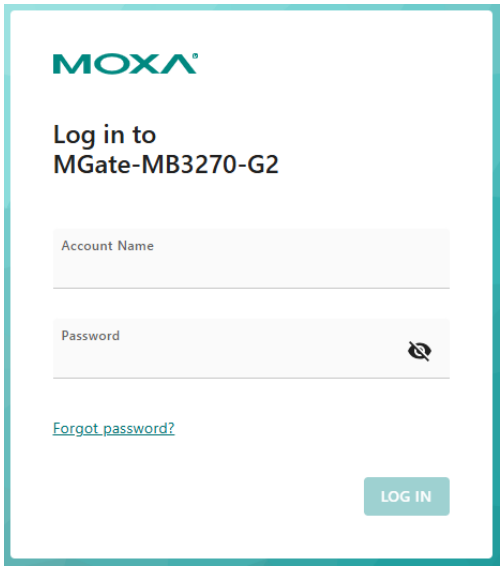
When you want to upgrade the firmware, select **Maintenance > Firmware Upgrade** and the **Choose File** button to find the firmware file. Select the **UPLOAD** button to proceed.



Ensure the device remains powered on and Select the **UPLOAD & RESTART** button. The device will upgrade to the new firmware version and restart itself.

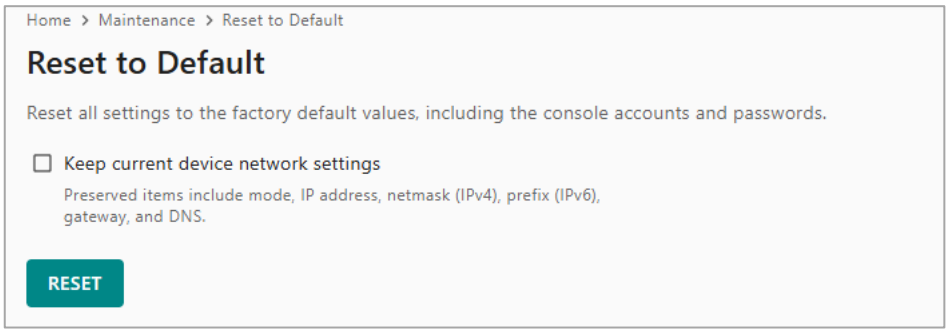


When the login page appears, it means that the firmware upgrade process has been completed.



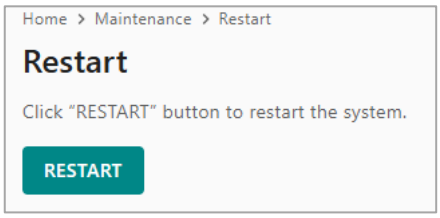
Reset to Default

This function will reset all the MGate's settings to the factory default values. All previous settings, including the console password, will be lost. If you wish to keep the MGate MB3000-G2 IP address, netmask, and other network settings, make sure **Keep current device network settings** is checked before loading the factory defaults.



Restart

If you want to restart the device, access **Maintenance > Restart** and select the **RESTART** button. The device will restart immediately.



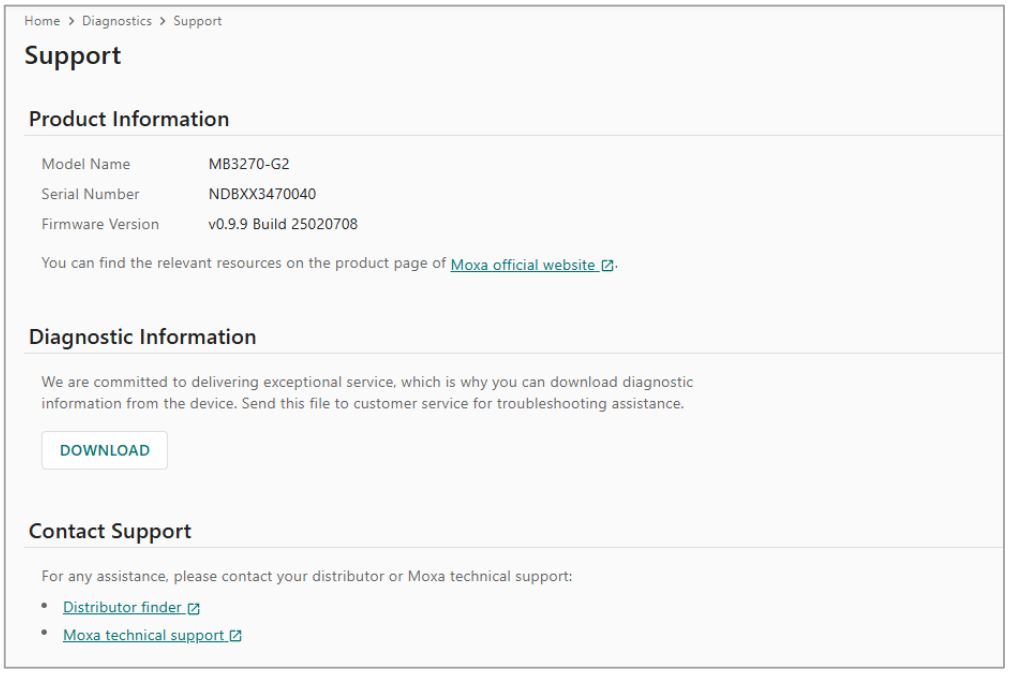
Diagnostics

System integrators and technical engineers may encounter issues when configuring a new application or receiving error reports during system operation. When that happens, you might find it helpful to have some diagnostic tools for troubleshooting.

In the Navigation Panel, the Diagnostics section brings together all the necessary functions for quick troubleshooting.

Support

If you need direct support from Moxa, go to **Diagnostics > Support** page. There, we provide relevant information before contacting Moxa, as well as contact information.



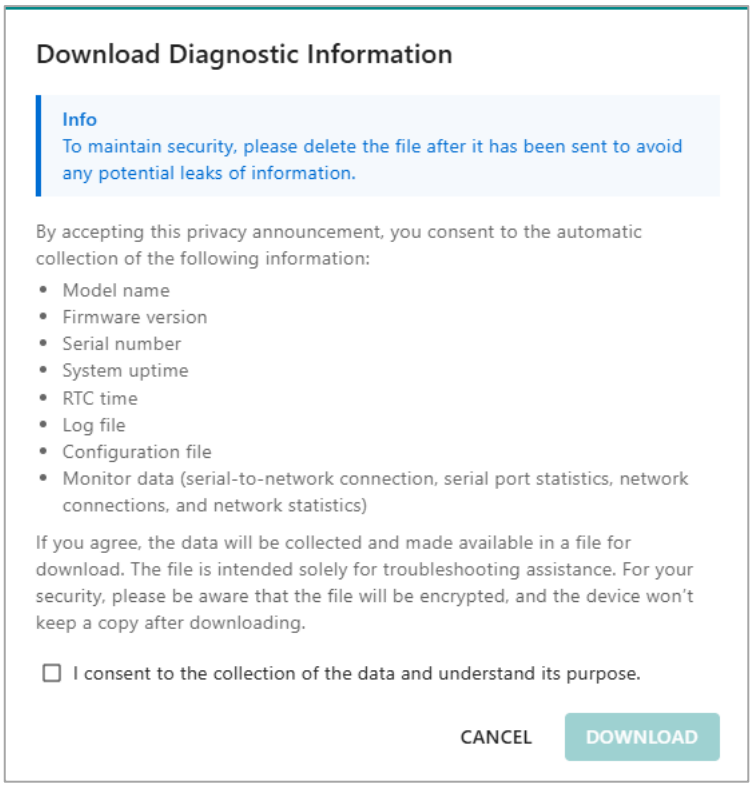
Product Information

Find here the basic information of the MGate protocol gateway, including the Model Name, Serial Number, Firmware Version of the MGate device.

Diagnostic Information

Previously, users would typically reach out to Moxa customer service initially, and the engineer would then request additional information for problem analysis. For the MGate MB3000-G2 Series, we advise users to gather diagnostic information and send it along with their inquiry to Moxa's customer service. This can make it simpler for the customer service engineer to pinpoint the root cause of the problem.

Select **DOWNLOAD** to save the data after providing your consent for collection.



The Download Diagnostic Information window will open and list what information on the MGate will be collected/downloaded. The diagnostic information is encrypted to ensure it is secure when delivered on the Internet and can only be unzipped by Moxa engineers for troubleshooting purposes. Access will not be granted with the password.

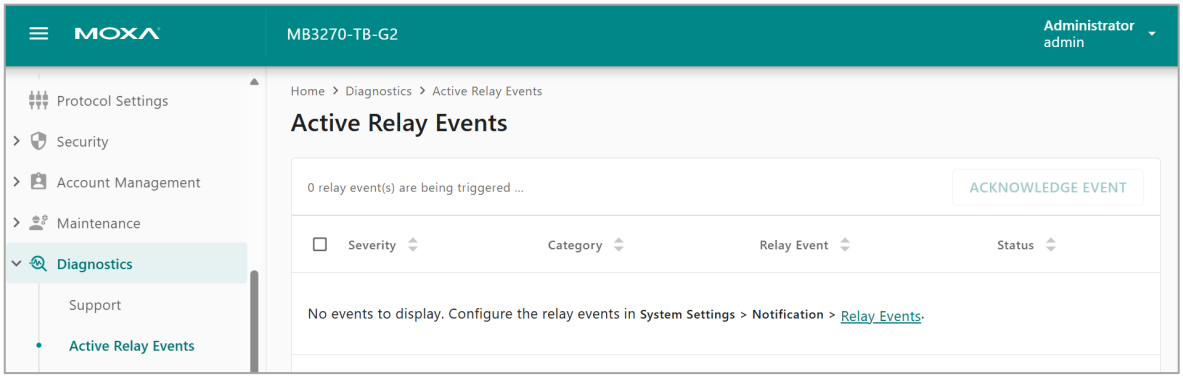
To get MGate's device information related to troubleshooting, use the web console.

Contact Support

After downloading the Diagnostic Information, you can find the contact window by selecting the **Distributor finder** or **Moxa technical support**, which will guide you to the corresponding resources on the official website.

Active Relay Events

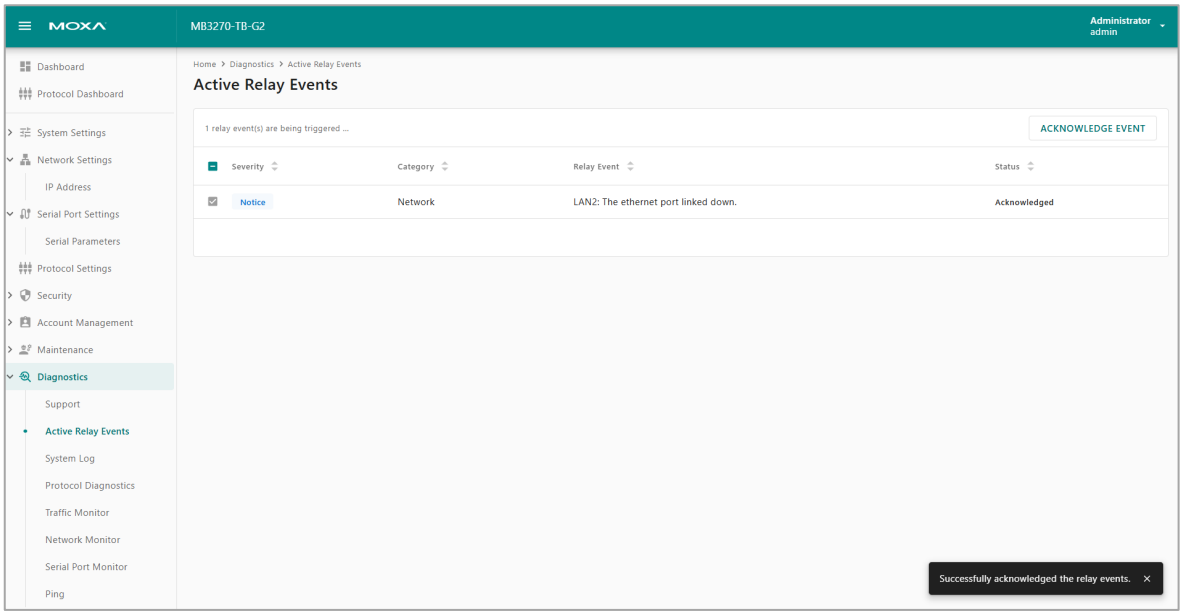
The main function of a relay output is to control the on/off state of a circuit, which can be connected to an alarm system or devices. For instance, when a "Power off" or "Ethernet link off" event occurs on MGate, it triggers the relay, causing the beeper to sound an alarm or the LED to light up, thereby indicating that a problem has occurred. Relay events can be configured under **System Settings > Notification > Relay Event**.



If a configured alarm event occurs and the MGate's relay output status changes, this page will show that events have occurred. If you want to clear the event temporarily, select on **ACKNOWLEDGE EVENT**.

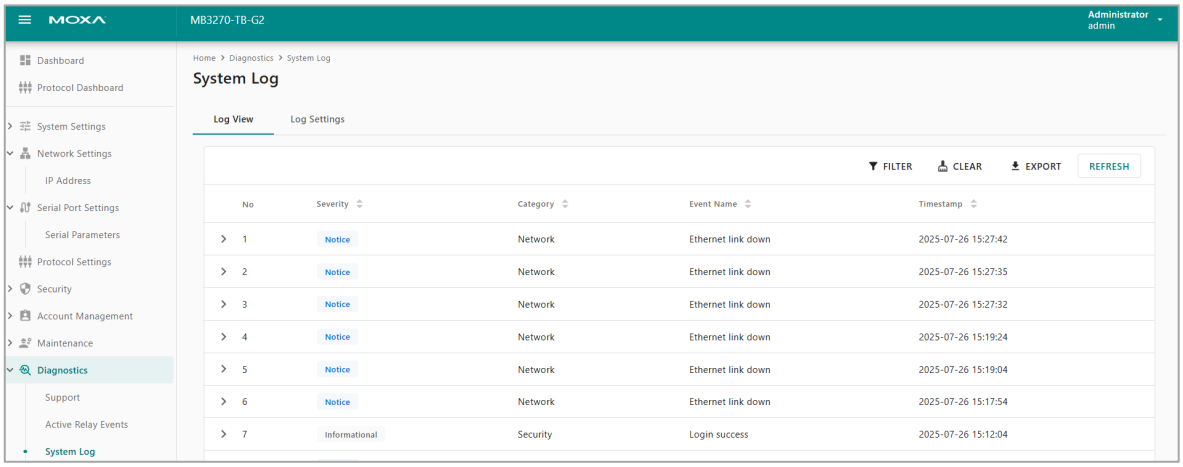
NOTE

The alarm message will remain on the web UI until the actual event or issue has been resolved.



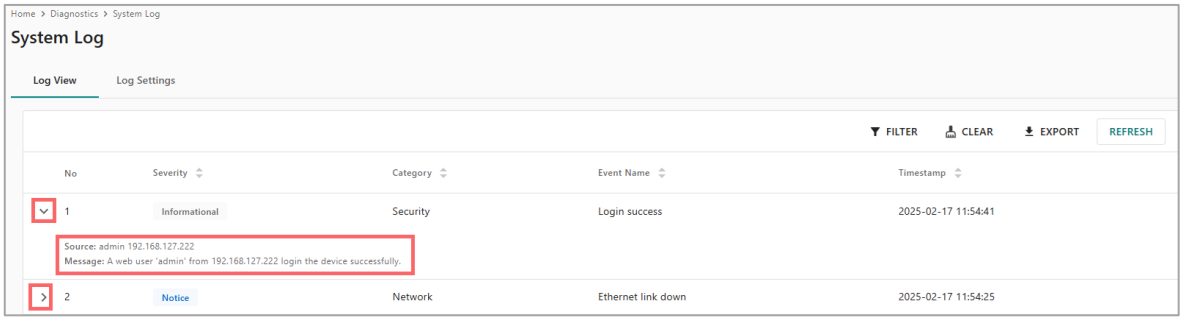
System Log

It is very important to record the activities of a device. At the System Settings > Notification page, configure which events will be recorded. Under the **Diagnostics > System Log > Log View** tab, find the recorded events on the MGate. Under the **Log Settings** tab, set the advanced settings for the local system log.

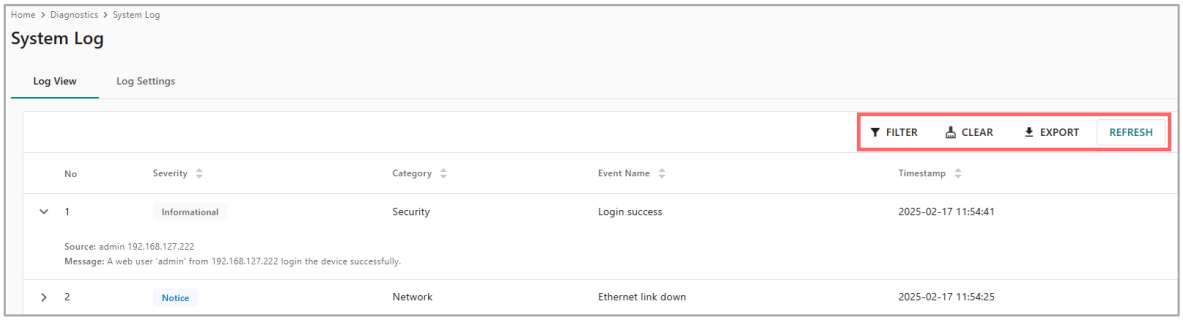


An event will be recorded under these columns: Severity, Category, Event Name, and Timestamp. Configure notification events to notify users actively at **System Settings > Notification** section. The detailed event list is in the appendix of this user manual.

Select the arrow icon to read more details about the event.



The MGate provides a filter, clear log, export log to file, and refresh to help read the events easily.



The detailed function is described below.

FILTER: Filter the event by Severity, Category, Event Name, or Timestamp.

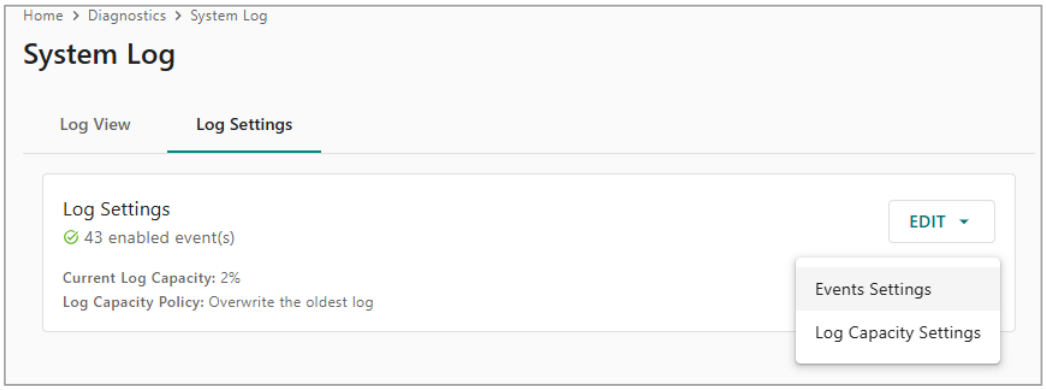
CLEAR: Delete all system logs on the device.

EXPORT: Export the system log for troubleshooting.

REFRESH: Refresh the logs on the panel.

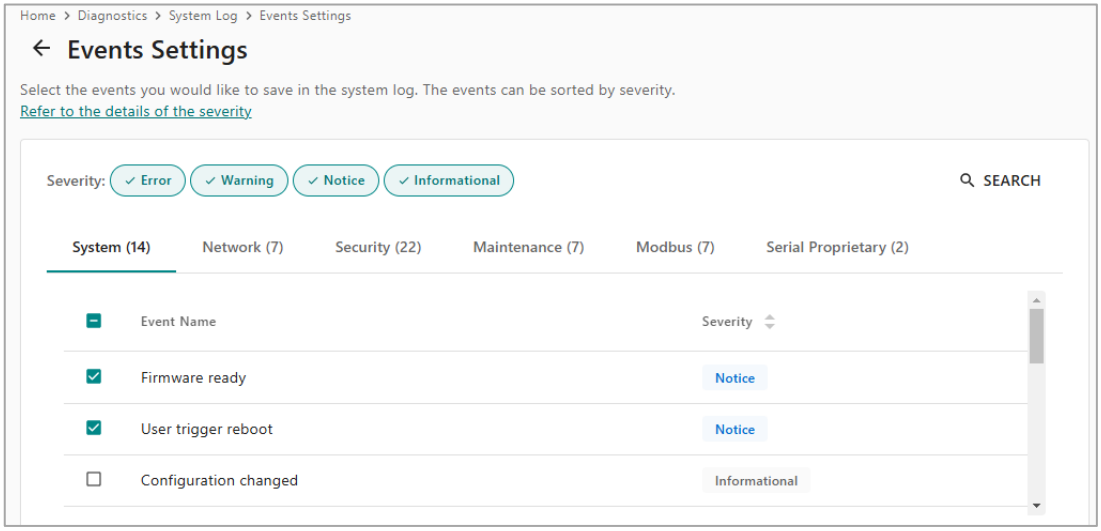
Under the **Log Settings** tab, you will see the Current Log Capacity displayed as a percentage for reference.

Since the events are stored on the local flash memory, there is a limitation on the number of events that can be saved. Select the **EDIT** button to manage the settings.



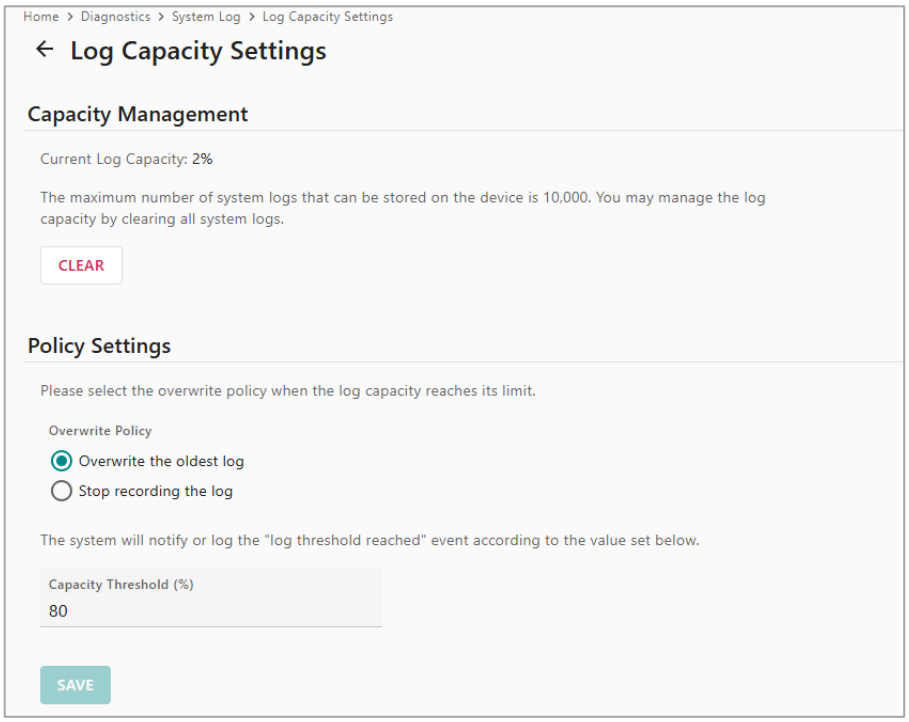
Events Settings

After selecting the **EDIT** button under **Log Settings > Event Settings**, you can select the events you would like to show in the local system log (the logs stored in MGate).



Log Capacity Settings

After selecting the **EDIT** button under Log Settings > **Log Capacity Settings**, configure what to do when the logs exceed the maximum size or clear the logs immediately.



Capacity Management: The MGate provides 10,000 audit records. Select the CLEAR button to clear the local system log when it's getting full.

Policy Settings: When the log capacity reaches its limit, decide what action the MGate should take because of the limited recording system log capacity.

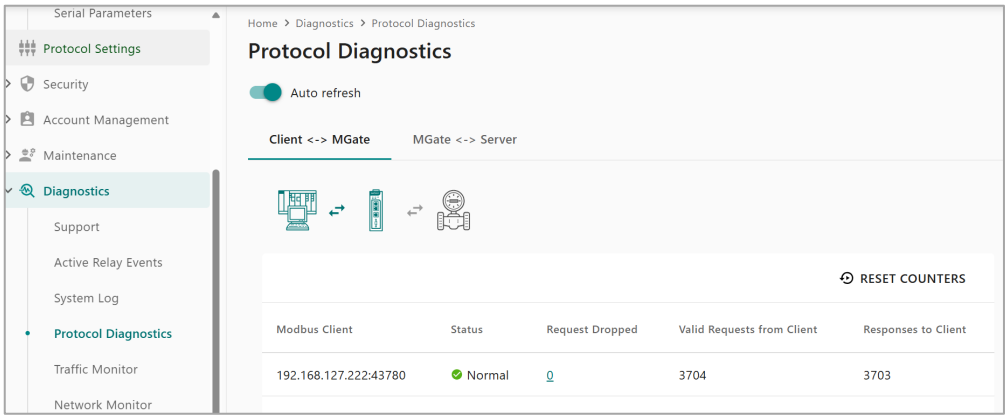
- Overwrite the oldest event log
- Stop recording events

Capacity Threshold (%): The system will notify you or record an event "log threshold reached" when the log capacity reaches the value set here. The default value is 80.

Protocol Diagnostics

If you have issues with the protocol conversion through MGate, the MGate provides easy-to-understand diagnostics information. The Protocol Diagnostics page is different when using **Transparent Mode** and **Agent Mode** due to their characteristics.

In **Transparent Mode**, as Modbus commands are directly converted and passed through, the protocol diagnostics page shows the communication statistics between Modbus client and MGate, and between the MGate and the Modbus server devices. Switch between tabs to quickly check the communication of both sides.



The diagnostics statistics description is shown below:

Connected Modbus TCP Client

Displays all connected TCP client device information and communication statistics with each client device. Individual information includes:

- **Modbus Client:** The IP:TCP port of the connected Modbus TCP client.
- **Status** (updated based on the latest communication results with connected devices):
  - **Normal:**
    - Connection successful
    - Communication normal
  - **Error:** Route to destination failed: check your topology and Modbus routing settings under Transparent Mode.
  - **Warning:**
    - Server queue buffer full: MGate can queue 32 requests for each Modbus client. Check your communication status.
    - Duplicate request before the response is received: The MGate hasn't received a response for the previous request, and receives the same request again.
- **Responses to client:** The counts of responses from MGate to Modbus client.
- **Valid Requests from client:** The counts of requests from Modbus client to MGate.
- **Dropped Requests:** Total counts of the following
  - Data format error
  - Server queue buffer full
  - Route to destination failed
  - Duplicate request before the response is received

Connected Modbus TCP Server

Displays all TCP server device information, statuses, and communication statistics. Individual information includes:

- **Modbus server:** IP/TCP Port of connected Modbus TCP servers.
- **Status** (updated based on the latest communication result with connected devices):
  - **Not Started:** Connection established but no communication yet.
  - **Normal:** Connection and communication is normal.
  - **Communication Failed:** Timeout for the last sent request.
  - **Connection Failed**
- **Requests to Server:** The number of requests sent from MGate to the server.
- **Valid Responses from server:** The count of responses sent from server to the MGate.
- **Dropped Responses:** Total counts of the following
  - Data format error
  - Unexpected responses
  - Unknown responses
  - Expired frame
- **Server Exceptions:** The counts of the exception received from the server devices.
- **Response Timeouts:** The number of response timeouts occurred.

Connected Modbus Serial Client

Displays communication statistics for all Modbus serial client devices.

- **Modbus Client:** Serial Port Number 1~4.
- **Status** (updated based on the latest communication result):
  - **Unknown:** No serial devices connected or no communication yet.
  - **Normal:** Communication normal.
  - **Error:** Route to destination failed. Check your topology and the Modbus routing settings under Transparent Mode.
  - **Warning:**
    - Server queue buffer full: MGate can queue 32 requests for each Modbus client. Check your communication status.
    - Duplicate request before the response is received: The MGate hasn't received a response for the previous request and receives the same request again.
  - **Note:** When the serial connection is disconnected for a receiving role, the status cannot be determined or updated, so it will remain in the current state.
- **Sent Responses**
- **Valid Requests**
- **Dropped Requests** (total of the following counters):
  - Data format error
  - Server queue buffer full
  - Route to destination failed
  - Duplicate requests before the response is received

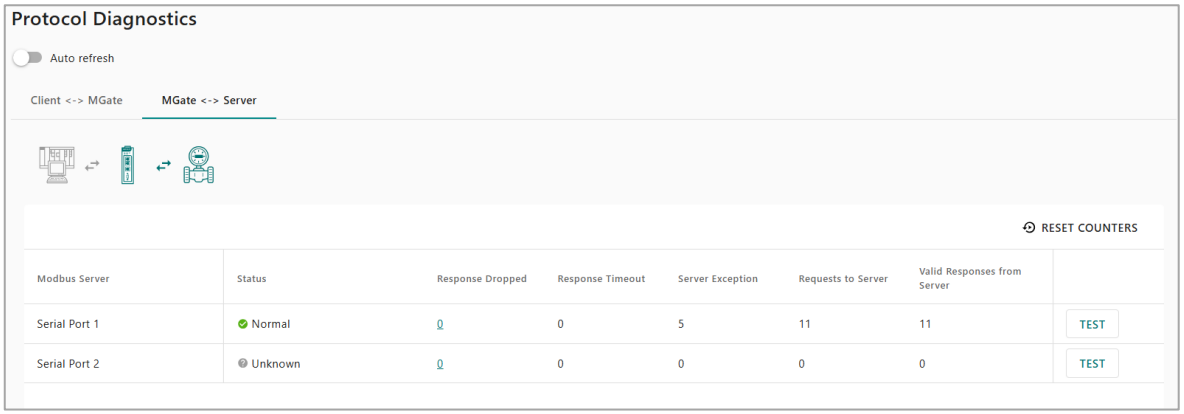
Connected Modbus Serial Server

Displays all serial server device statuses and communication statistics.

- **Modbus Server:** Serial Port Number 1~4.
- **Status** (updated based on the latest communication result):
  - **Unknown:** No serial devices connected or no communication yet.
  - **Normal:** Communication normal.
  - **Communication Failed:** Timeout for the last sent request.
- **Sent Requests:** The counts of sent requests to Modbus servers.
- **Valid Responses:** The count of received responses to MGate.
- **Dropped Responses** (total of the following counters):
  - Data format error
  - Unexpected responses
  - Unknown responses
  - Expired frame
- **Server Exception:** The connected server device responded with an exception.
- **Response Timeout:** The MGate passed requests to the server but the server did not respond in time.

Transparent Mode—Modbus Test Tool

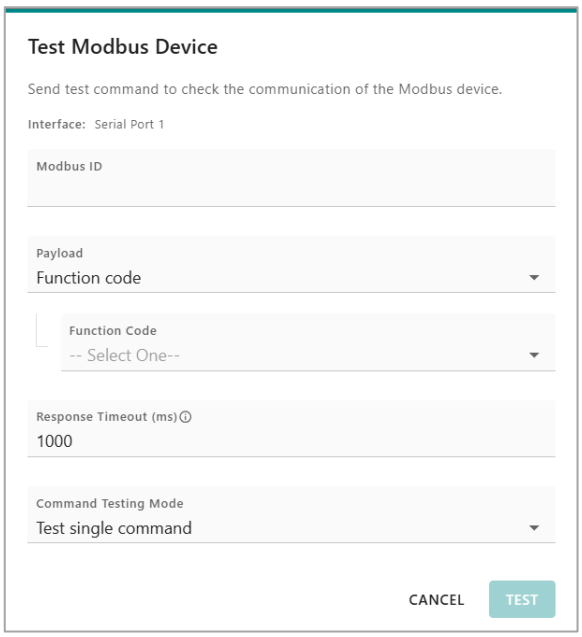
This is a very useful tool for troubleshooting in Transparent Mode. To test if the communication between MGate and the server device is fine, under the MGate <-> Server tab, you can select the **TEST** button to read or write a value directly to the Modbus or serial server. You don't need to install any additional tools and can easily and actively test the device from the MGate by using this feature.



NOTE

A connection to a Modbus TCP server will show only when the Modbus TCP client has initiated connection transparently with that Modbus TCP server through the MGate. It does not support creating a Modbus TCP server in the MGate transparent mode.

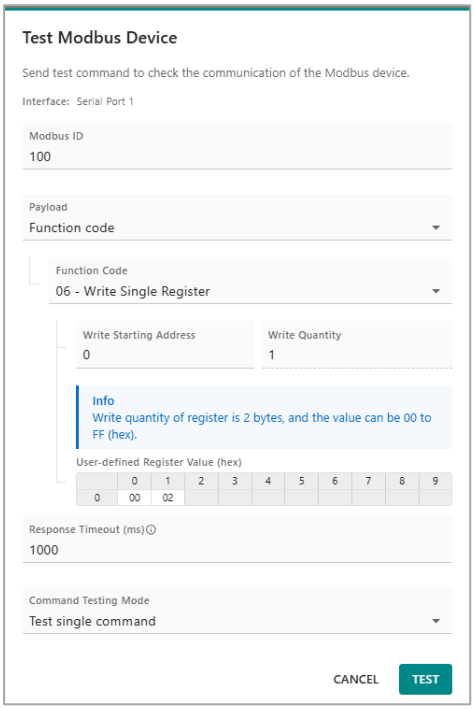
In the pop-up window, you can set up the following parameters to test:



Modbus ID: The Modbus ID that you want to test with

Payload:

- **Standard Function Code + payload**
  - Standard Modbus Read Function Code: 01/02/03/04, you can define the read address and quantity
  - Standard Modbus Write Function Code: 05/06/15/16, you can define the write address, quantity, and the data, like the following example:



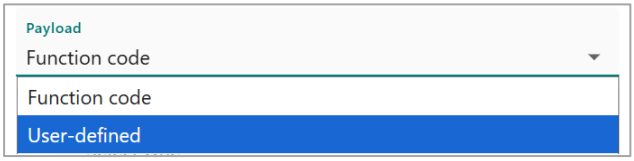
- **User-define payload:** Fill in the desired payload.

NOTE

User-defined payloads exclude Modbus ID, CRC, or LRC. The format should be in Hex; separated by spaces. When using Modbus RTU, the maximum length is **253 bytes**. When using Modbus ASCII, the maximum length is **506 bytes**.

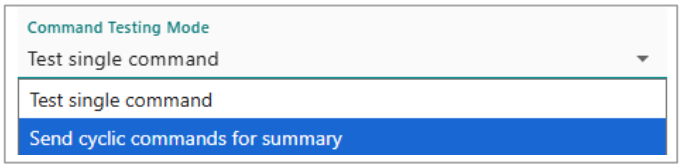
Example—Modbus RTU test payload: 0x 03 00 00 00 02

Example—Modbus ASCII test payload: 0x 30 33 30 30 30 30 30 30 30 32

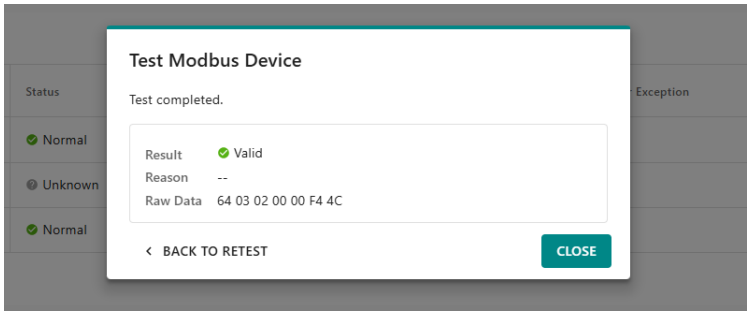


Response Timeout: Set up the Modbus response timeout in ms for the test command. If the target device does not respond within this timeout, MGate assumes the communication has failed.

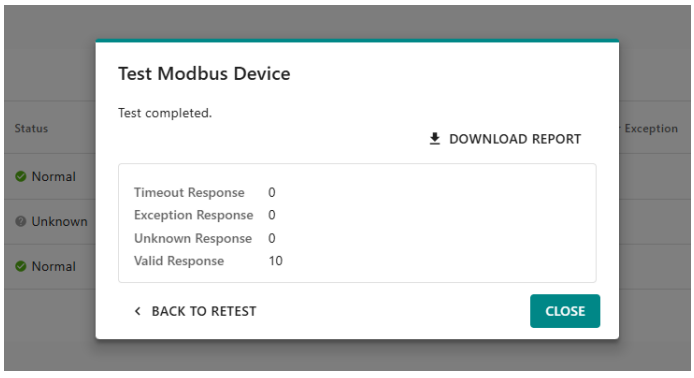
Command Testing Mode:



- Test Single Command: The MGate sends one command and returns the result of Return status, Exception/Unknown response Reason, and the responded Raw data.



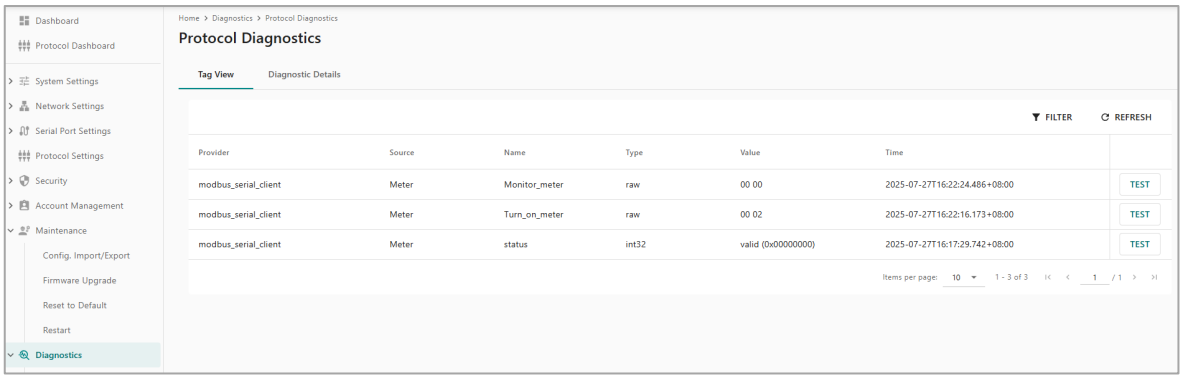
- Send cyclic Command for summary: You can configure the number of times and interval for MGate to send test commands. After cyclic testing, the MGate shows a summary of the test results. For test details and raw data of every command, download the text report.



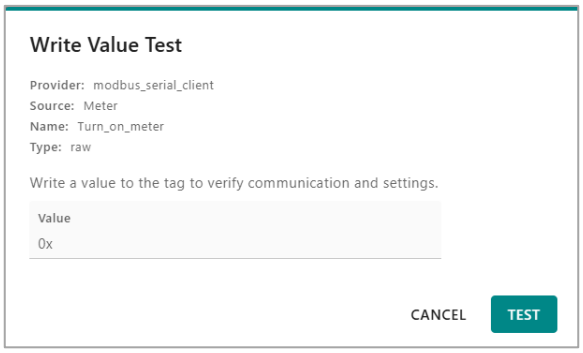
In **Agent Mode**, the MGate actively polls the end devices and caches the data in its memory, and different protocols operate independently. On the protocol diagnostics page, you can see two parts: the **Tag View** and the **Diagnostics Details**.

Agent Mode—Tag View and Test Tool

After you create tags under agent mode, this tab displays the live tag value generated by field devices and updates the values periodically. It is an easy and useful tool if you want to check whether the MGate receives the correct data from field devices. The gateway timestamp shows the time data was updated to the tag.

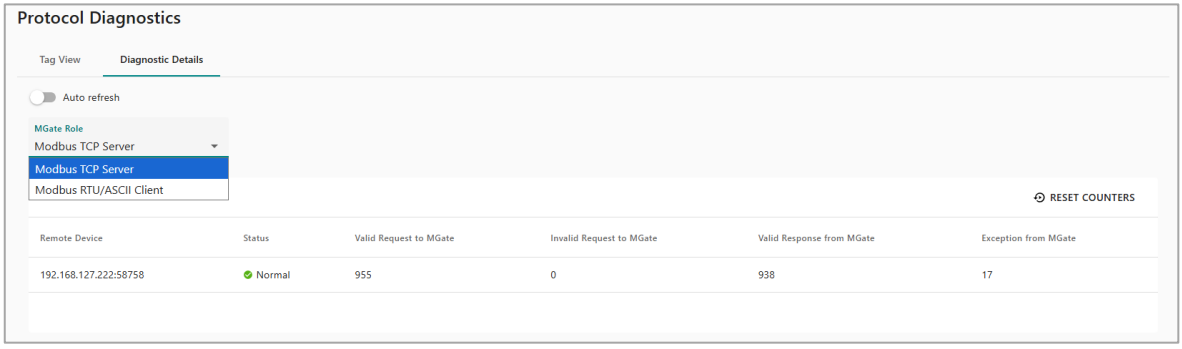


To test if the communication between the MGate and the device is fine, you can select the **TEST** button to write a value directly to the Modbus or serial server. You don't need to install any additional tools; you can actively test the device from the MGate by using this feature.



Agent Mode—Diagnostics Details

In this tab, you can check the overview of protocol communication status, communication statistics, including the remote device, valid or invalid requests and responses, exceptions sent from MGate.

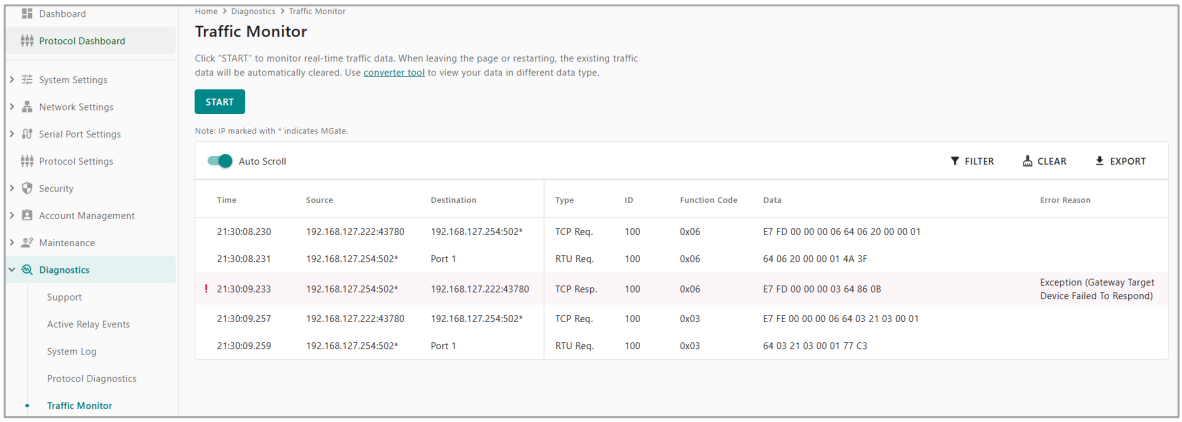


Based on your topology, you can select the desired MGate roles to check the diagnostics information of different protocols.

Traffic Monitor

For troubleshooting purposes, you can monitor the Modbus or serial data passing through the MGate. The MGate shows the traffic in an easy-to-understand format with interpreted fields, including the timestamp, source, destination, type, ID, Modbus Function Code, Data, and Error reason. Events can be filtered in different ways, and the complete log can be exported to a csv file for offline analysis.

Both the Modbus TCP and the Modbus serial/serial packets can be captured. To initiate capturing, select the **START** button; the transactions captured will be shown below.



In **Transparent Mode**, all Modbus commands are transparently converted and passed through MGate, so both Modbus TCP and Modbus serial side's traffic can be monitored on the same view.

See below for the Transparent Mode traffic log fields definition:

Time: The timestamp of this MGate device's current time, in the form of ISO 8601, and the format of hh:mm:ss.sss, which excludes year, month, and date. The seconds is precise to three decimal places.

Source: Depending on your topology, could be an IP address, Serial Port, or MGate itself.

Destination: Depending on your topology, could be an IP address, Serial Port, or MGate itself.

Type: Modbus TCP, RTU, or ASCII request/response.

ID: The Modbus ID, indicating the Modbus device that you are communicating with.

Function Code: The Modbus Function Code, such as 03 (Read Holding Registers), 06 (Write Single Register)

Data: The raw data value of data.

NOTE

All of the above data, including time, source, destination, type, ID, function code and raw data, is shown in Hex.

Error Reason:

Protocol	Error	Likely Cause	How to Troubleshoot
TCP	Unknown Packet (length < 8 bytes)	Incomplete frame, truncated packet, non-Modbus TCP traffic	Check network stability, ensure client uses correct Modbus framing
	Error PID	Protocol ID ≠ 0x0000 (non-Modbus TCP traffic)	Verify only Modbus TCP clients connect to the ethernet port
	Error Length (header length ≤ data length < expected)	MBAP length field doesn't match actual data received	Monitor for packet loss or fragmentation from Modbus TCP
	Exceed Maximum Length (> 261 bytes)	Incorrect, oversized or corrupted/collided frames	Check client application to confirm frame size

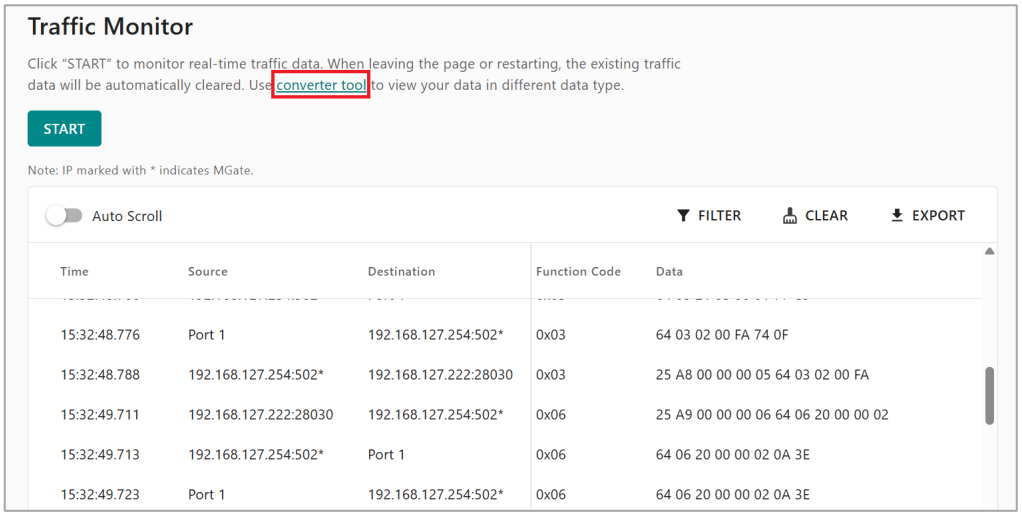
Protocol	Error	Likely Cause	How to Troubleshoot
RTU	Error Length (< expected length)	Missing bytes because of noise, wrong Modbus timeout, misaligned serial settings	Check wiring, verify serial settings, verify if traffic is truncated, adjust Modbus timeout settings
	Short frame for Function Code 08 or unknown code	Special case – minimum frame not met	Validate client request format
	CRC Error	Frame corruption (noise, EMI, wrong serial config, frame collision if multiple devices send frames at the same time)	Check shielding/grounding, verify serial parameters, inspect cabling for interference, check if multiple clients or servers are sending Modbus frames on the same serial bus
	Exceed Maximum Length (> 256 bytes)	Malformed or corrupted frame	Verify client data length, confirm the server's response size, check if multiple clients or servers are sending Modbus frames on the same serial bus
ASCII	Unknown Packet (length < 9 bytes)	Incomplete frame	Ensure client constructs full ASCII frame, check for dropped bytes
	Start Character Error	First byte not ':'	Verify client encoding (must start with colon)
	End Character Error	Missing or invalid CR LF	Check client library for correct termination
	LRC Error	Frame corruption or miscalculated LRC	Validate client checksum logic, check for transmission errors
	Exceed Maximum Length (> 513 bytes)	Oversized or invalid frame	Verify client PDU construction, check for protocol misuse

After finishing the capturing, you have the option to select the **FILTER** button to narrow down the data for analysis or select the **EXPORT** button to save the transactions for further analysis by Moxa customer service. Select **CLEAR** to clear the traffic log captured.

NOTE

When you switch to other MGate configuration pages or the session times out when the traffic log is running, the Traffic Monitor will be stopped.

In Transparent Mode MGate also provides a useful built-in Converter Tool for users to easily convert the Modbus raw data into other formats, so you can compare with your device's value and check if they match.

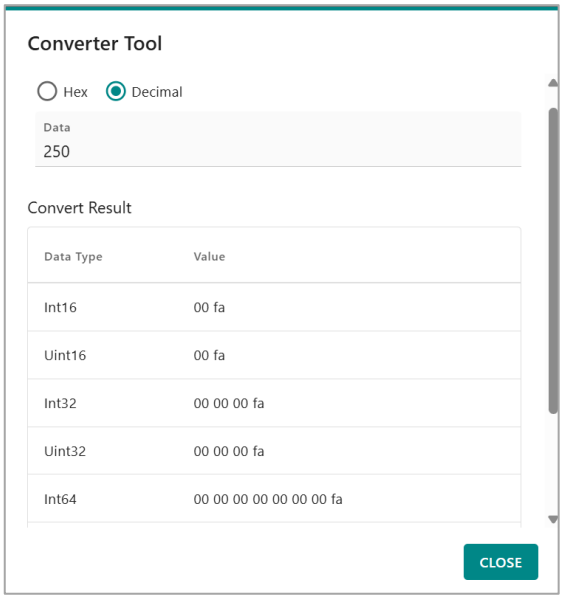


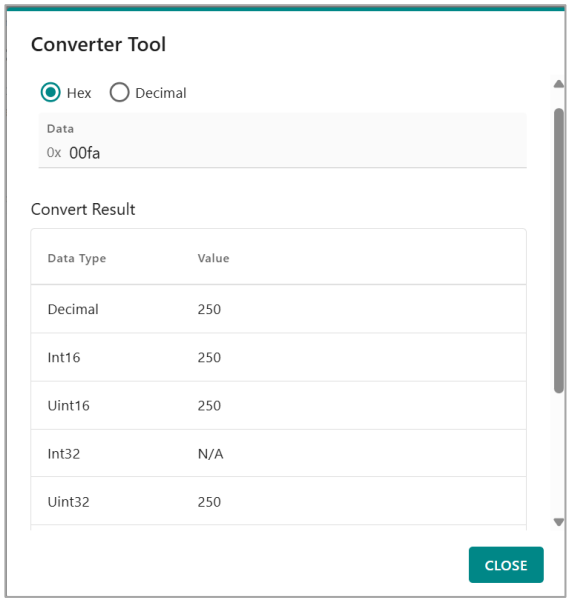
Input Hex or Decimal raw values of the data you want to check, usually it is the "Data" field from the traffic log, part of the Data field, or the real values on your client/server device. The tool can convert Hexadecimal or Decimal data into Int16, Uint16, Int32, Uint32, Float, and Double.

See an example below:

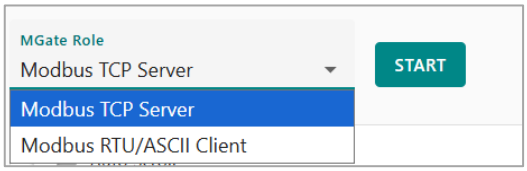
The value shown on the real meter is 250 (Hz); however, you see an issue on the Modbus client, so you are not sure about whether the MGate is passing the correct values.

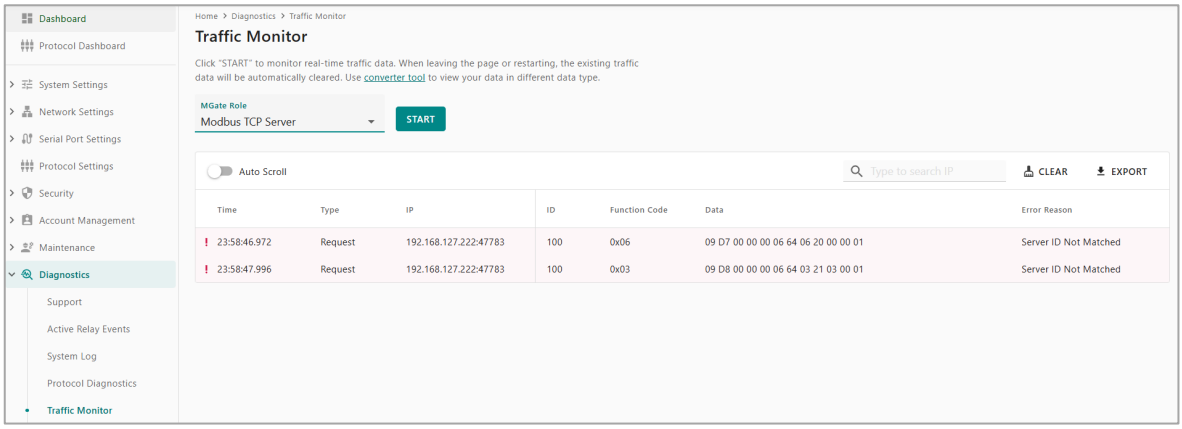
Check in either way: input the real value on your device/SCADA (e.g. 250) to the tool to see if the Modbus data matches, or copy the Modbus data part (e.g. 00fa, in Hex) and paste it to the tool, you can find that the values match. Here there might be some other setting issues on the client that need to be troubleshot.





In **Agent Mode**, there will be two protocol roles on the MGate that are working independently. For example, if the topology is Modbus TCP client to Modbus RTU server, then the MGate's role would be Modbus TCP server and Modbus RTU client. Select the role to decide which side's protocol traffic you'd like to monitor.





See below for the Agent Mode traffic log field definitions:

When the MGate acts in a Modbus TCP or serial server role:

Time: The timestamp of this MGate device's current time, in the form of ISO 8601, and the format of hh:mm:ss.sss, which excludes year, month, and date. The seconds are precise to three decimal places.

Type: Modbus Request or Modbus Response:

- "Request": The Modbus server (MGate) received a normal response
- "Response": The response sent from the Modbus server (MGate) to the client
- "Request/Invalid": The Modbus server (MGate) received invalid frames (e.g. incomplete frames, CRC/LRC error when in Modbus serial)
- "Request/Not Supported": The Modbus server (MGate) received Modbus function code that is not supported.
- "Request/Server ID Not Matched": The Modbus server (MGate) received a Modbus ID that is not matched to the Modbus request.

IP: IP address and port (only shows in Modbus TCP only)

Serial Port: serial port number (only shows in Modbus serial only)

Type: "RTU" / "ASCII" (only shows in Modbus serial only)

ID: The Modbus ID.

Function Code: The Modbus Function Code, shown in decimal.

Data: The Modbus raw data.

When the MGate acts as a Modbus TCP or serial client role:

Time: The timestamp of this MGate device's current time, in the form of ISO 8601, and the format of hh:mm:ss.sss, which excludes year, month, and date. The seconds are precise to three decimal places.

Type: Modbus Request or Modbus Response:

- "Request": The request sent from the Modbus client (MGate)
- "Response": The response sent to the Modbus client (MGate)
- "Request/Resend": When a response timeout occurs on the Modbus server, the Modbus client re-sends request.
- "Response/Invalid": The Modbus client (MGate) received invalid frames (e.g. incomplete frames, CRC/LRC error when in Modbus serial)
- "Response/Unknown": The Modbus TCP client (MGate) received a response with a different Transaction Identifier. The Modbus serial client received responses with a different function code or Modbus ID than the request.

IP: IP address and port (only shows in Modbus TCP)

Serial Port: Serial port number (only shows in Modbus serial)

Type: "RTU" / "ASCII" (only shows in Modbus serial only)

ID: The Modbus ID.

Function Code: The Modbus Function Code, shown in decimal.

Data: The Modbus raw data.

When the MGate acts in a serial proprietary client role:

Time: The timestamp of this MGate device's current time, in the form of ISO 8601, and the format of hh:mm:ss.sss, which excludes year, month, and date. The seconds are precise to three decimal places.

Type: Request/Response/Produce/Consume, details below:

- "Request": The request sent by proprietary Serial/ASCII client (MGate)
- "Response": The response is received by proprietary Serial/ASCII client (MGate)
- "Request/Resend": The proprietary Serial/ASCII client (MGate) resends requests, e.g. The client (MGate) detects a server response timeout and resends the request.
- "Response/Invalid": The proprietary Serial/ASCII client (MGate) received an invalid frame (e.g. incomplete frame, constant error or not matched with the settings, checksum check error)
- "Response/Unknown": The proprietary Serial/ASCII client (MGate) received a complete frame, but the response list is empty. Please check your proprietary serial settings under protocol settings.
- "Produce": The proprietary Serial/ASCII client (MGate) sends produce commands.
- "Consume": The proprietary Serial/ASCII client (MGate) received normal consume commands.
- "Consume/Invalid": The proprietary Serial/ASCII client received an invalid frame (e.g. incomplete frame, constant error or not matched with the settings, checksum error)
- "Consume/Unknown": The proprietary Serial/ASCII client received a complete frame, but the consume list is empty. Please check your proprietary serial settings under protocol settings.

Serial Port: 1to 4 port.

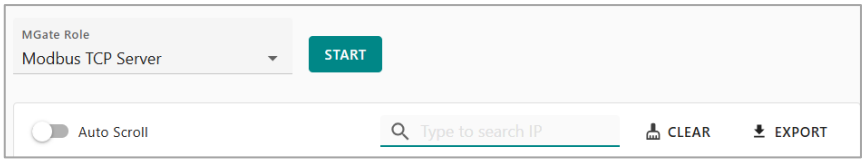
Data Length: The data size in bytes.

Data: The Modbus raw data.

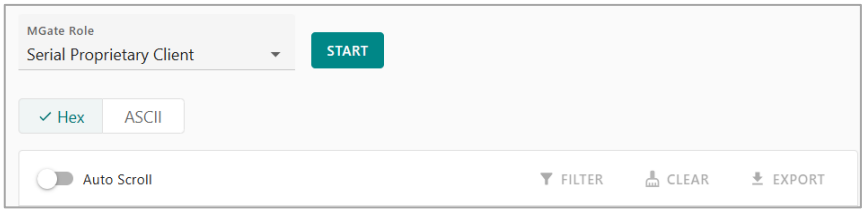
After finishing the capturing, you have the option to use the search field under Modbus TCP to search for specific IP addresses or select the **FILTER** button under Modbus serial or proprietary serial to narrow down the data for analysis. You can also select the **EXPORT** button to save the transactions for further analysis by Moxa customer service. Select **CLEAR** to clear the traffic log captured.

NOTE

If you switch to other MGate configuration pages or the session times out while the traffic log runs, the system will stop the Traffic Monitor.

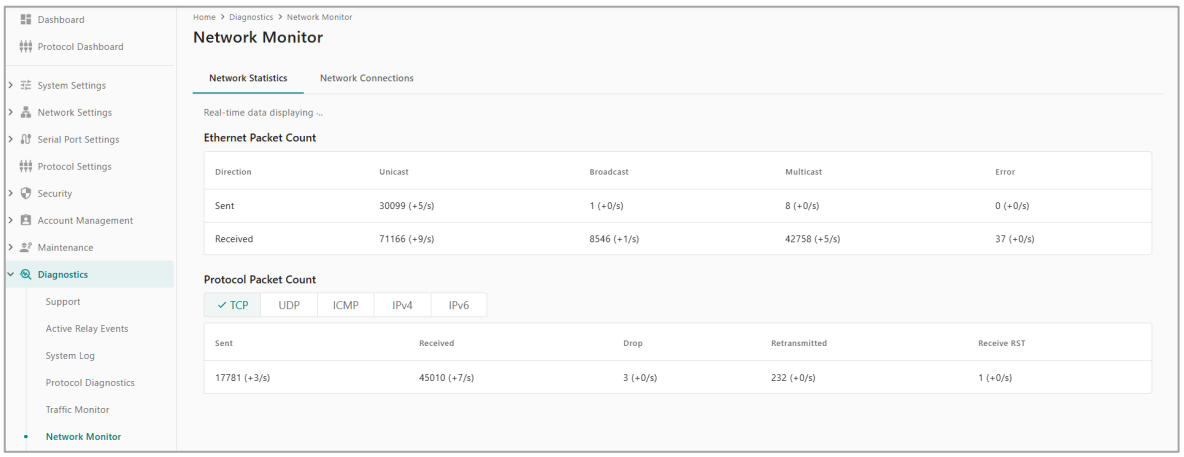


For serial proprietary, you can use the tab to view data in Hex or ASCII format.



Network Monitor

The MGate is a protocol gateway, so users usually only need to diagnose communication errors at the Modbus level. However, if sometimes you need to check the Ethernet status, for example, the current connection status to the MGate, or IP packets status. Everything that happens on the Ethernet interface will be recorded here, **Diagnostics > Network Monitor**, to help you understand the Ethernet data transmitted/received.



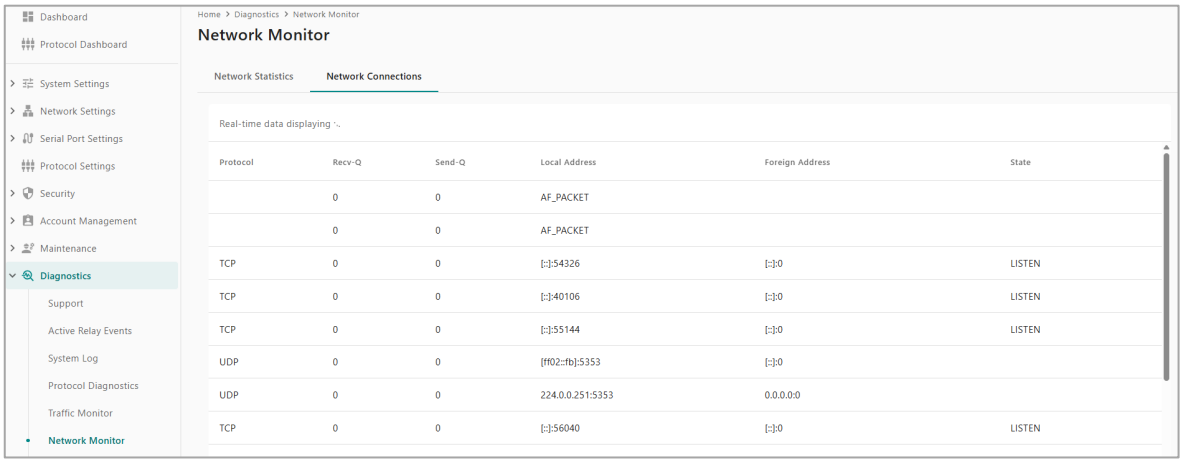
Network Statistics

The Ethernet Packet Count sheet separates the Ethernet data in two directions, Send and Received, to count the number of unicasts, broadcasts, and multicasts. If there are any error bytes, the Error column will count them.

The Protocol Packet Count sheet separates the Ethernet data by different protocols to count the numbers of TCP, UDP, ICMP, IPv4, IPv6, and PPP.

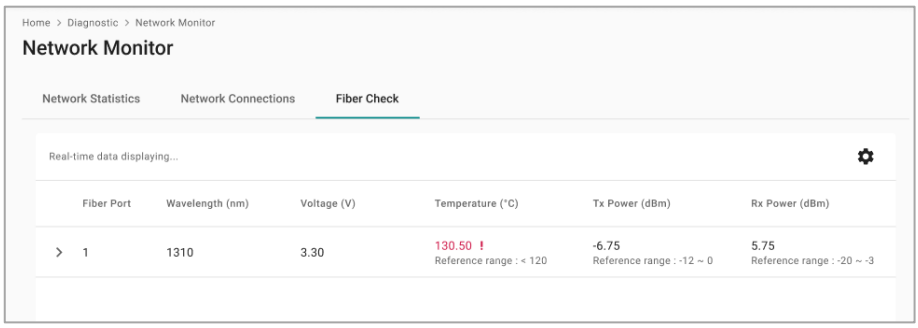
Network Connections

This tab displays the status of all the TCP sessions.

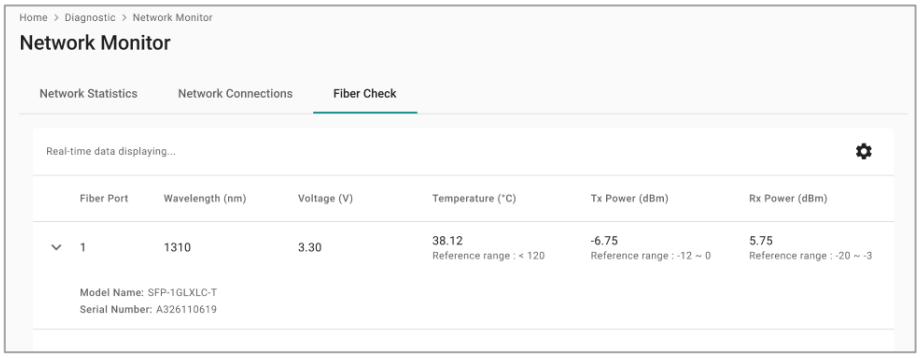


Fiber Check

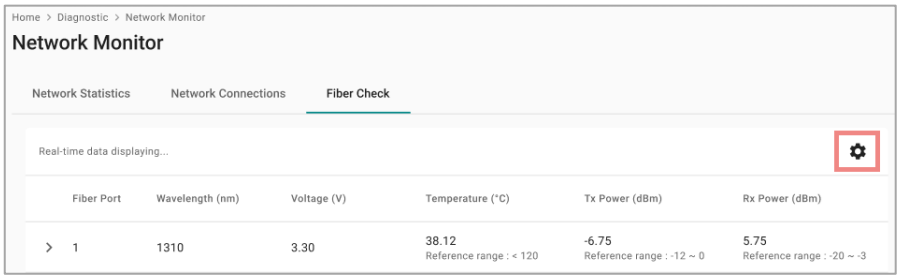
If you are using a Fiber model, including SFP or fixed type (Multi-mode SC/ST & Single-mode SC), you will see the Fiber Check tab on the Network Monitor page. Fiber Check is used to diagnose the link status of fiber connectors, monitor the module's temperature, TX/RX power, and other parameters on fiber ports to determine if the ports are working properly.

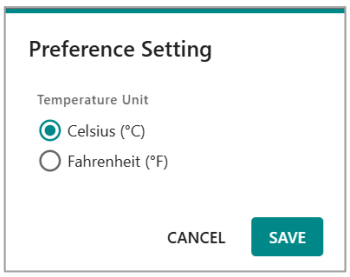


By selecting the arrow, you can expand the detailed fiber module information.



You can also select the settings button to change the temperature units.





For SFP MGate models, only Moxa SFP modules support full Fiber Check functionality. The communication may work using third-party modules, but the Fiber Check function may provide incomplete data. The supported Moxa SFP modules are as below:

Model Name	Temperature Threshold (°C)	Tx Power (Max./Min.) (dBm)	Rx Power (Max./Min.) (dBm)
FEMST (Fixed type fiber)	120	-14/-20	-3.0/-32.0
FEMSC (Fixed type fiber)	120	-14/-20	-3.0/-32.0
FESSC (Fixed type fiber)	120	0.0/-5.0	-3.0/-34.0
SFP-1FEMLC-T (separate SFP module)	120	-8.0/-18.0	-3.0/-32.0
SFP-1FESLC-T (separate SFP module)	120	0.0/-5.0	-3.0/-34.0
SFP-1FELLC-T (separate SFP module)	120	0.0/-5.0	-3.0/-34.0

Serial Port Monitor

The Serial Port Monitor shows the physical status of each serial port. The MGate is a protocol gateway, so users usually only need to diagnose communication errors at the Modbus level. However, if sometimes you need to check the physical serial status, for example, to determine if the serial link is off or some serial hardware settings might be incorrect, check this page.

Serial Tx: The serial data (in bytes) that are sent out of MGate's serial port.

Serial Rx: The serial data (in bytes) that are received on MGate's serial port.

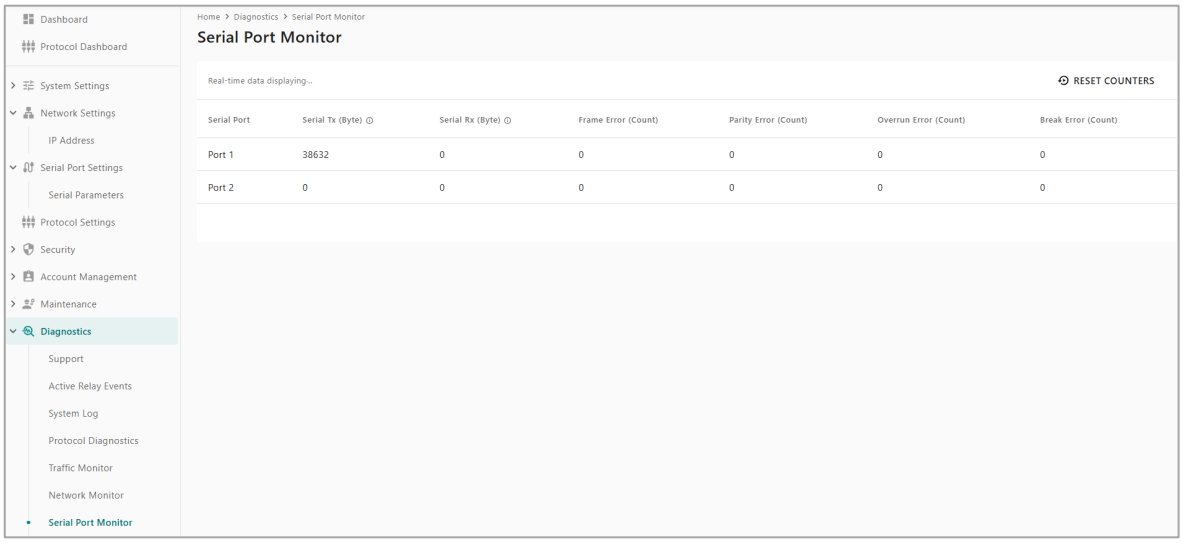
Frame: Framing error shows that the received character did not have a valid stop bit.

Parity: Parity error shows that the received data character does not match the parity selected.

Overrun: The MGate cannot hand-receive data to a hardware buffer because the input rate exceeds the MGate's ability to handle the data.

Break: Break interrupt shows that the received data input was held low for longer than a full-word transmission time. A full-word transmission time is defined as the total time to transmit the start, data, parity, and stop bits.

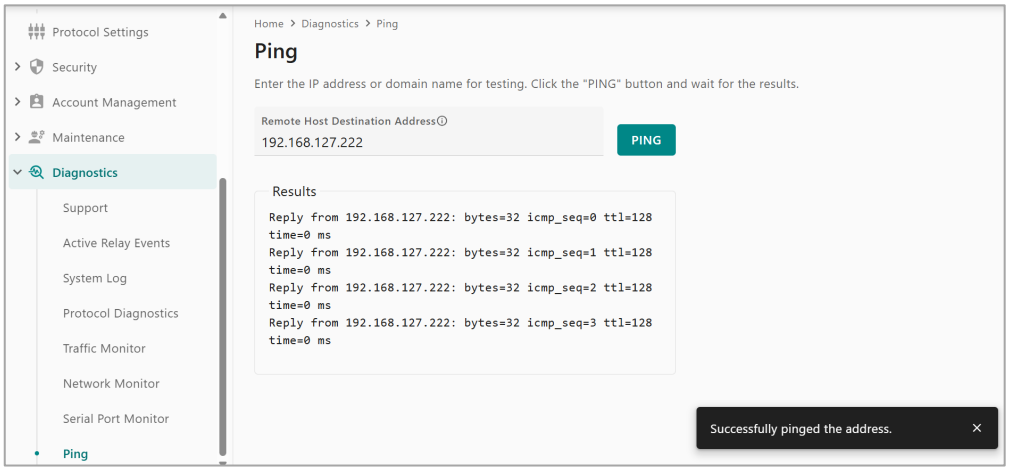
To clear the counters for troubleshooting, select **RESET COUNTERS**.



Ping

The Ping function is a good tool for troubleshooting. Use this tool to verify the status of network nodes connected to MGate.

Directly input the IP address and select the **PING** button. The MGate will check if the target node can respond to the ping request and display the result.



9. Mass Deployment/Maintenance

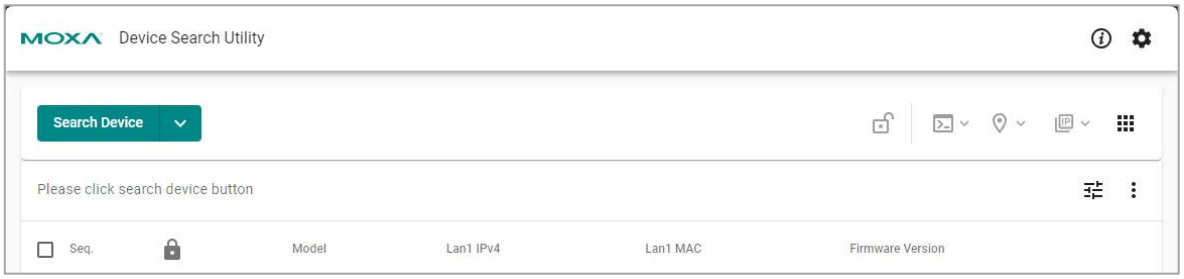
Once a user completes the settings on a device server, they may need to deploy those settings to multiple devices or sites. Moxa provides the GUI tool Device Search Utility v3.0 or the CLI tool Moxa CLI Command Tool, MCC Tool, to meet this requirement.

After the devices were set up at the locations, the maintainer might need to perform routine tasks on a regular basis to run the system. This includes tasks such as firmware upgrades or password updates. The Device Search Utility v3.0 and MCC Tool can assist the maintainer in carrying out these tasks effortlessly.

Mass Configuration With GUI Tool: Device Search Utility v3.0 or Newer

The Device Search Utility v3.0 is a web-based utility. Make sure the operating system and browser version comply with the below versions before using the tool:

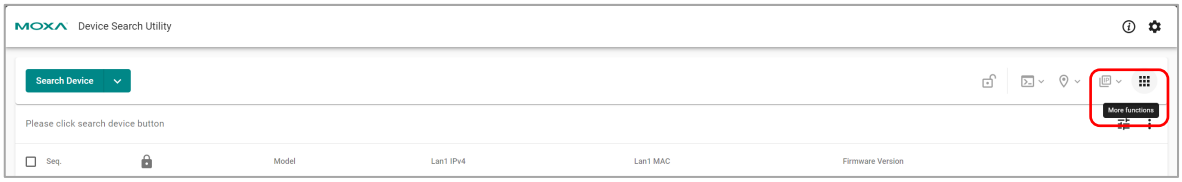
- Chrome:
  - For Windows 7, 8/8.1, Server 2012 and Server 2012 R2: Chrome 109 and newer
  - For Windows 10 and newer, Server 2016 and newer: All Chrome versions
- Firefox:
  - For Windows 7 and newer versions, Server 2012 and newer versions: All Firefox ESR versions
- Edge:
  - For Windows 7 and newer versions, Server 2012 and newer versions: All Firefox ESR versions



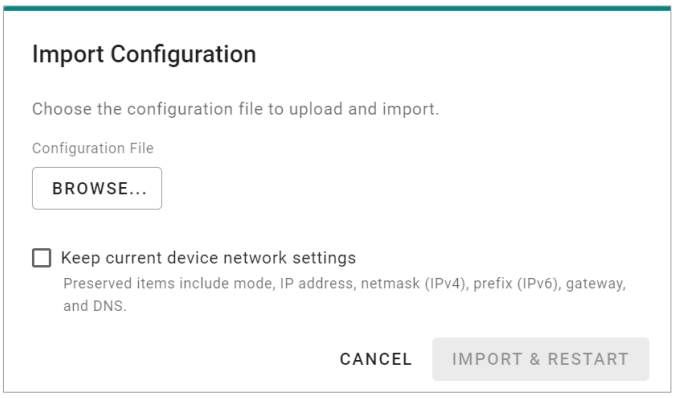
Execute the Device Search Utility and select the Search Device button to find the target MGate(s). Remember to unlock them before any further actions.

Import/Export Configuration

Select the MGate(s) to import/export configuration and then move the mouse to the More functions to choose the Import Configuration function.



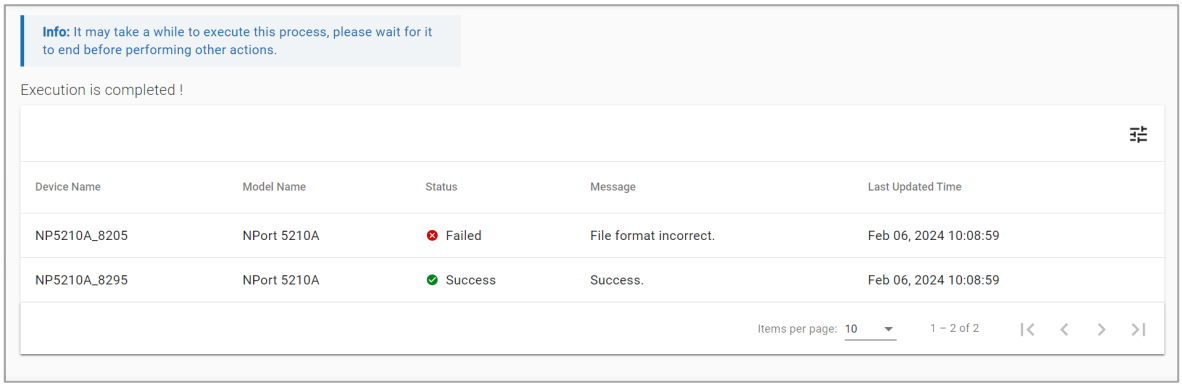
Import Configuration is to import one configuration file to one or more devices with the same model. Select the BROWSER… button to find out where the configuration file is saved.



Keep the Current Device Network Settings

If the target MGate(s) already has the proper IP address(es) configured, you may choose to retain the existing network settings for the device(s). Select the option.

After importing the configuration, Device Search Utility will display success or failure in the Status & Message columns for each device.



Your device may restart again to make the configuration effective, and it will stop your work in progress.

NOTE

To troubleshoot the cause of failure, refer to the *DSU User Manual Appendix: Error Messages*.

For exporting the configuration file(s), you can also find the Export Configuration function under the More functions button.

Export Configuration is to export the configuration file from one or more devices with the same model. When exporting one device only, the file format may be *.ini, *.dat, *.txt, *.cfg, *.dec. The filename will be [ModelName] - [IP] _ [Date] .xxx, e.g., MGate MB3170-10.123.10.1_220724.ini.

When exporting multiple devices, the system will zip the configuration files.

Import Certificate

To build a more secure or a zero-trust network environment, you may want to set up a public key infrastructure (PKI). The certificate needs to be imported onto all network devices for this scenario. To simplify the loading process, the Device Search Utility supports importing certificates to multiple MGates.

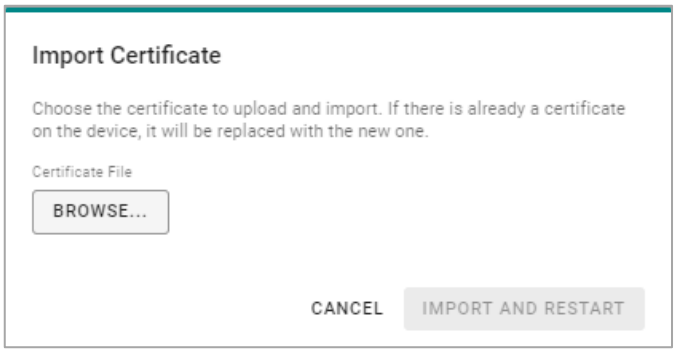
Find the Import Certificate function under the More functions button.

Import Certificate is to exchange certificate files to one or more devices to establish secured command/data transferring.

Step 1: Select the MGate MB3000-G2 models

Step 2: Import certificate file

Step 3: Import and restart



Firmware Upgrade

The increasing convergence of IT and OT poses a cybersecurity risk as more OT network devices connect to office networks. Upgrading the firmware version to the latest one is crucial for all network devices. To meet this requirement, the Device Search Utility supports firmware updates on multiple MGates.

You can find the Firmware Upgrade function under the More functions button. **Firmware Upgrade** is to send one firmware file to one or more devices with the same model. The firmware file extension normally ends in .ROM.

Step 1: Select MGate MB3000-G2 models

Step 2: Import firmware file

Step 3: Imported and the device will restart.

Mass Configuration With CLI Tool: MCC Tool

The MCC Tool is a command-line utility based on the Windows and Linux platforms. Make sure you have downloaded the correct file for your operating system.

Unzip the file and install the MCC Tool. Execute the MCC Tool under the command line to manage the MGates in the network.

Import/Export Configuration

Import/Export the device configuration for a specific device or a range of devices through the device list file. The password must be specified by the parameter or by the device list file. Device configurations are stored in individual files, using device type, IP address, and file create date as the filename. The result log is directly printed on the screen, or you can specify a result_log file to store the result.

MCC_Tool –cfg –ex –i [ip_address] –u [user] –p [password] –dk [key] –l [result_log]

MCC_Tool –cfg –ex –d [Device_list] –l [result_log]

MCC_Tool –cfg –ex –d [Device_list] –l [result_log] –t [timeout_value]

MCC_Tool –cfg –im –i [ip_address] –u [user] –p [password] –dk [key] –f [cfg_file] –l [result_log] –n –nr

MCC_Tool –cfg –im –d [Device_list] –l [result_log] –n –nr

MCC_Tool –cfg –im –d [Device_list] –l [result_log] –t [timeout_value]

Parameters Description:

Command	Function	Remark
-cfg	Execute actions related to configuration	
-ex	Export the configuration file	
-im	Import the configuration file	
-i	Device IP address (ex. 192.168.1.1)	
-d	Device list	
-u	Device's user account for login	
-p	Device's password for login	
-dk	**When Exporting configuration:** The command decrypts the exported file with the pre-shared key. • If this parameter is not used, the exported file will be encrypted by the pre-shared key set on the firmware of the device. • If this parameter is used, the exported file will be decrypted to a clear-txt file for editing. **When Importing configuration:** If the configuration file that needs to be imported is encrypted, the command is needed with pre-shared key. • If the import configuration file is without -n, The MCC tool will ignore -dk (won't return -11). • If the import configuration file is with - n, the MCC tool will use pre-shared key to decrypt the encrypted file. Therefore, if the key is wrong for decrypting the file, the MCC tool will return -10. However, if the file is in plain text, and you input the pre-shared key, it will ignore the key (won't return -10).* (by parameter -dk or the key column in the device list file)	
-f	The configuration file to be imported	Only for the import configuration function
-n	Keep original network parameters (includes IP, subnet mask, gateway, and DNS)	Only for the import configuration function
-nr	Do not reboot the device after importing the configuration file	Only for the import configuration function.
-l	Export result log file	
-t	Timeout (1 to 120 seconds) Export function Default value: 30 seconds Import function Default value: 60 seconds	

Example: Export the configuration using a device list and export the results to a result log

MCC_Tool –cfg –ex –d [DeviceList] –l [result_log]

The result_log will include the following items:

Model	ServerName	IP	MAC	FwVer	CfgFile	Key	ErrCode
MGate_MB3170;	MGate_MB3170_123;	192.168.1.1;	00:90:e8:01:02:03;	1.3;	MB3170_192_168_1_1_20170622.ini	moxa;	0;
MGate_MB3170;	MGate_MB3170_456;	192.168.1.2;	00:90:e8:04:05:06;	1.3;	MB3170_192_168_1_2_20170622.ini	moxa;	0;

Example: Import the configuration to a device list (with restarting the units) and export the results to a result log.

MCC_Tool –cfg –im –d [DeviceList] –l [result_log]

The result_log will include the items below:

Model	ServerName	IP	MAC	FwVer	CfgFile	Key	ErrCode
MGate_MB3170;	MGate_MB3170_123;	192.168.1.1;	00:90:e8:01:02:03;	1.3;	MB3170_192_168_1_1_20170622.ini	moxa;	0;
MGate_MB3170;	MGate_MB3170_456;	192.168.1.2;	00:90:e8:04:05:06;	1.3;	MB3170_192_168_1_2_20170622.ini	moxa;	0;

Example: Import the configuration to a device list without restarting the units and export the results to a result log.

MCC_Tool –cfg –im –d [DeviceList] –nr –l [result_log]

Firmware Upgrade

With the IT/OT convergence trend, office networks may see an increase in OT network devices, posing cybersecurity risks. Upgrading the firmware version is crucial for all network devices. The MCC Tool allows users familiar with the command-line interface to update the firmware on multiple MGates to fulfill this need.

The MGate MB3000-G2 Series supports password protection by default and cannot be disabled. The password(s) must be specified by a command parameter or by the DeviceList file before upgrading the firmware and restarting a specific device (or multiple devices simultaneously).

MCC_Tool –fw –up –i [ip_address] –u [user] –p [password] –f [firmware_file] –l [result_log]

MCC_Tool –fw –up –d [Device_list] –l [result_log]

MCC_Tool –fw –up –d [Device_list] –l [result_log] –t [timeout_value]

Parameters Description:

Command	Function	Remark
-fw	Execute actions related to firmware	
-up	Upgrade firmware version	
-i	Device’s IP address (192.168.1.1)	
-u	Device’s user account for login	
-p	Device’s password for login	
-d	Device list	
-f	Firmware file to be upgraded	
-l	Export result log file	
-t	Timeout (1~1200 seconds) Default value: 800 seconds	
-print	Print upgrade process status message	

Example: Upgrade firmware using a device list and capture the results in an import log.

MCC_Tool –fw –u –d [DeviceList] –l [result_log]

The result_log will include the items below:

Model	ServerName	IP	MAC	FwFile	ErrCode
MGate_MB3170;	MGate_MB3170_123;	192.168.1.1;	00:90:e8:01:02:03;	MB3170_V1.3.rom;	0;
MGate_MB3170;	MGate_MB3170_456;	192.168.1.2;	00:90:e8:04:05:06;	MB3170_V1.3.rom;	0;

Change Password

Because of the IT/OT convergence trend, an increasing number of companies require their employees to update their login passwords regularly, as do the network devices. The owner/maintainer of the network devices may need to update the password regularly. The MCC Tool helps you to ease this routine job by generating a small script to update the password.

Set the password of the target device specified by an IP address. The current password must be specified by a parameter or by the Device List file.

MCC_Tool –pw –ch –i [ip_address] –u [user] –p [old_password] –npw [new_password]

MCC_Tool –pw –ch –d [Device_list] –nd [device_list_new_password] –l [result_log]

MCC_Tool –pw –ch –d [Device_list] –nd [device_list_new_password] –l [result_log] –t [timeout_value]

Parameters Description:

Command	Function	Remark
-pw	Execute actions for password	
-ch	Change password	
-npw	The new password for a specific user	
-i	Device's IP address (192.168.1.1)	
-u	Device's user account for login	
-p	Device's password for login (old password)	
-d	Device list	
-nd	The Device list with new password settings	This command requires assigning a new password in the Device List.
-l	Export result log file	
-nr	Don't reboot the device after changing the password	
-t	Timeout (1to120 seconds) Default value: 60 seconds	

Example: Set the new password as "5678" and restart the device to make it effective. Print the result on the screen.

MCC_Tool –pw 5678 –i 192.168.1.1 –u admin –p moxa

Example: Set the new password from a device list and then restart the device to make it effective. Export the results to a result log

MCC_Tool –pw DeviceList_New –d [DeviceList] –l [result_log]

The result_log will include the items below:

Model	ServerName	IP	MAC	FwVer	User	PWD	ErrCode
MGate_MB3170;	MGate_MB3170_123;	192.168.1.1;	00:90:e8:01:02:03;	1.3;	admin;	5678;	0;
MGate_MB3170;	MGate_MB3170_456;	192.168.1.2;	00:90:e8:04:05:06;	1.3;	admin;	moxa;	0;

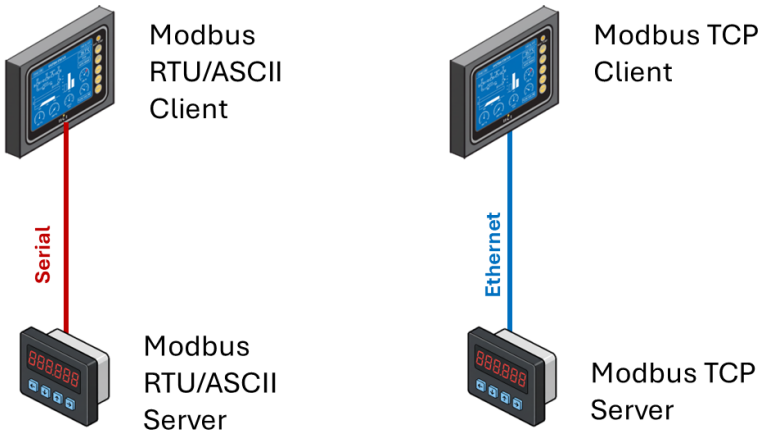
A. Modbus Overview

Introduction

Modbus is one of the most popular automation protocols in the world. It supports both serial and Ethernet devices. Many industrial devices, such as PLCs, DCSs, HMIs, instruments, meters, motors, and drivers, use Modbus as their communication standard.

Devices Are Either Clients or Servers

All Modbus devices are classified as either a client or a server. Clients start all communication with servers and do not communicate with other clients. Servers are completely passive and communicate only by sending a response to a client's request.



Servers Are Identified by ID

Each Modbus server in a system is assigned a unique ID between 1 and 247. Whenever a client makes a request, the request must include the ID of the intended recipient. Client devices themselves have no ID.

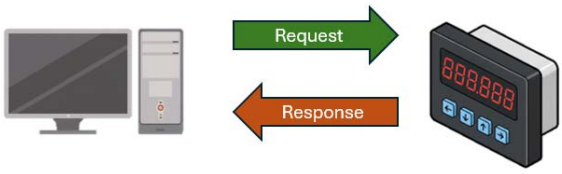
0	1 to 247	248 to 255
Broadcast address	Individual address of the server	Reserved

Communication Is by Request and Response

All Modbus communication is by request and response. A client sends a request, and a server sends a response. The client will wait for the server's response before sending the next request. For broadcast commands, no response is expected. These three scenarios are explained below:

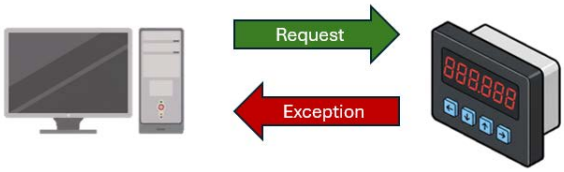
Normal

The client sends a request to the server. The server sends a response with the requested information.



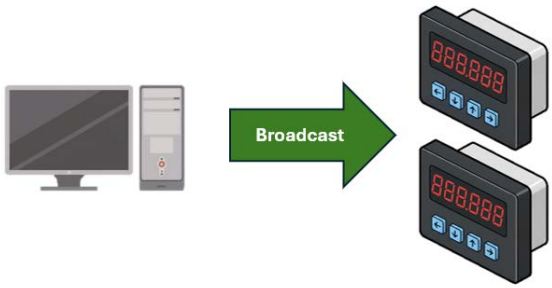
Exception

The client sends a request to the server. The server may not support the command, or an error is detected, so it sends an exception to the client.



Broadcast

The client sends a broadcast command, such as a reset command. Every server on the network complies with the command, and no response is sent to the client.



Requests Need a Time Limit

The original Modbus protocol was not designed for simultaneous requests or simultaneous clients, so only one request on the network can be handled at a time. When a client sends a request to a server, no other communication may be started until after the server responds. The Modbus protocol specifies that clients use a response time-out function to identify when a server is nonresponsive because of device or line failure. This function lets a client abandon a request after a certain time without a response. The following illustrates this.

Response Timeout

The client sends a request. The server is unresponsive for the time specified by the response timeout function. The client gives up on the request and resumes operation, allowing another request to be initiated.

To accommodate a wide range of devices, baudrates, and line conditions, manufacturers can determine actual response timeout values. This allows the Modbus protocol to accommodate a wide range of devices and systems. However, this also makes it difficult for system integrators to know what response timeout value to use during configuration, especially with older or proprietary devices.

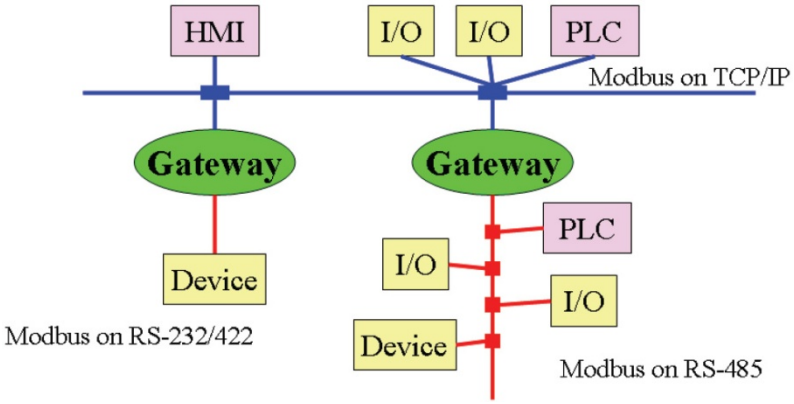
The MGate MB3000-G2 provides an auto-detection function that tests all attached devices and recommends a response timeout value. This function saves considerable time and effort for system integrators, and it results in more accurate timeout settings.

Modbus Ethernet vs. Modbus Serial

Although Modbus is intended as an application-layer messaging protocol, the data format, and communication rules for Ethernet-based Modbus TCP differ from serial-based Modbus ASCII and RTU.

The major difference between the Ethernet and serial Modbus protocols is in the behavior of the communication model. Modbus ASCII and RTU allow only one request on the network at a time. Once a request is sent, no other communication on the bus is allowed until the server sends a response, or until the request times out. However, Modbus TCP allows simultaneous requests on the network, from multiple clients to multiple servers. TCP clients cannot send more than one request at a time to a server, but they can send requests to other servers before a response is received. The Modbus TCP standard recommends that servers be able to queue up to 16 requests at a time. The MGate MB3000-G2 will queue up to 32 requests from each TCP client, for up to 16 or 32 TCP clients.

Integrate Modbus Serial and Ethernet With Gateways



Ordinarily, Modbus TCP and Modbus ASCII/RTU cannot communicate with each other. However, with a Modbus gateway between the Modbus serial network and the Modbus Ethernet network, TCP clients can communicate with serial servers and serial clients can communicate with TCP servers.

B. SNMP MIB List

The MGate MB3000-G2 Series has built-in Simple Network Management Protocol (SNMP) agent software that supports SNMP Trap, RFC1317 and RS-232-like groups, and RFC 1213 MIB-II.

RFC1213 MIB-II Supported SNMP Variables

System MIB	Interfaces MIB	IP MIB	ICMP MIB
sysDescr	ifNumber	ipForwarding	icmpInMsgs
sysObjectID	ifIndex	ipDefaultTTL	icmpInErrors
sysUpTime	ifDescr	ipInReceives	icmpInDestUnreachs
sysContact	ifType	ipInHdrErrors	icmpInTimeExcds
sysName	ifMtu	ipInAddrErrors	icmpInParmProbs
sysLocation	ifSpeed	ipForwDatagrams	icmpInSrcQuenchs
sysServices	ifPhysAddress	ipInUnknownProtos	icmpInRedirects
	ifAdminStatus	ipInDiscards	icmpInEchos
	ifOperStatus	ipInDelivers	icmpInEchoReps
	ifLastChange	ipOutRequests	icmpInTimestamps
	ifInOctets	ipOutDiscards	icmpTimestampReps
	ifInUcastPkts	ipOutNoRoutes	icmpInAddrMasks
	ifInNUcastPkts	ipReasmTimeout	icmpInAddrMaskReps
	ifInDiscards	ipReasmReqds	icmpOutMsgs
	ifInErrors	ipReasmOKs	icmpOutErrors
	ifInUnknownProtos	ipReasmFails	icmpOutDestUnreachs
	ifOutOctets	ipFragOKs	icmpOutTimeExcds
	ifOutUcastPkts	ipFragFails	icmpOutParmProbs
	ifOutNUcastPkts	ipFragCreates	icmpOutSrcQuenchs
	ifOutDiscards	ipAdEntAddr	icmpOutRedirects
	ifOutErrors	ipAdEntIfIndex	icmpOutEchos
	ifOutQLen	ipAdEntNetMask	icmpOutEchoReps
	ifSpecific	ipAdEntBcastAddr	icmpOutTimestamps
		ipAdEntReasmMaxSize	icmpOutTimestampReps
		ipRouteDest	icmpOutAddrMasks
		ipRouteIfIndex	icmpOutAddrMaskReps
		ipRouteMetric1	
		ipRouteMetric2	
		ipRouteMetric3	
		ipRouteMetric4	
		ipRouteNextHop	
		ipRouteType	
		ipRouteProto	
		ipRouteAge	
		ipRouteMask	
		ipRouteMetric5	
		ipRouteInfo	
		ipNetToMediaIfIndex	
		ipNetToMediaPhysAddress	
		ipNetToMediaNetAddress	
		ipNetToMediaType	
		ipRoutingDiscards	

Address Translation MIB	TCP MIB	UDP MIB	SNMP MIB
atIfIndex	tcpRtoAlgorithm	udpInDatagrams	snmpInPkts
atPhysAddress	tcpRtoMin	udpNoPorts	snmpOutPkts
atNetAddress	tcpRtoMax	udpInErrors	snmpInBadVersions
	tcpMaxConn	udpOutDatagrams	snmpInBadCommunityNames
	tcpActiveOpens	udpLocalAddress	snmpInBadCommunityUses
	tcpPassiveOpens	udpLocalPort	snmpInASNParseErrs
	tcpAttemptFails		snmpInTooBigs
	tcpEstabResets		snmpInNoSuchNames
	tcpCurrEstab		snmpInBadValues
	tcpInSegs		snmpInReadOnlys
	tcpOutSegs		snmpInGenErrs
	tcpRetransSegs		snmpInTotalReqVars
	tcpConnState		snmpInTotalSetVars
	tcpConnLocalAddress		snmpInGetRequests
	tcpConnLocalPort		snmpInGetNexts
	tcpConnRemAddress		snmpInSetRequests
	tcpConnRemPort		snmpInGetResponses
	tcpInErrs		snmpInTraps
	tcpOutRsts		snmpOutTooBigs
			snmpOutNoSuchNames
			snmpOutBadValues
			snmpOutGenErrs
			snmpOutGetRequests
			snmpOutGetNexts
			snmpOutSetRequests
			snmpOutGetResponses
			snmpOutTraps
			snmpEnableAuthenTraps
			snmpSilentDrops
			snmpProxyDrops

RFC1317 RS-232-like Groups

RS-232 MIB	Async Port MIB
rs232Number	rs232AsyncPortIndex
rs232PortIndex	rs232AsyncPortBits
rs232PortType	rs232AsyncPortStopBits
rs232PortInSigNumber	rs232AsyncPortParity
rs232PortOutSigNumber	
rs232PortInSpeed	
rs232PortOutSpeed	

Input Signal MIB	Output Signal MIB
rs232InSigPortIndex	rs232OutSigPortIndex
rs232InSigName	rs232OutSigName
rs232InSigState	rs232OutSigState

C. Event List

The MGate MB3000-G2 provides event logs to help users troubleshoot. All the events that may be recorded are listed below.

Item	Category	Severity	Default Setting	Event Name	Description
1	System	Notice	Disable	Firmware ready	The system is ready for operation.
2		Notice	Disable	User trigger reboot	The device was rebooted by the user.
3		Informational	Disable	Configuration changed	A user changed the configuration setting, and the new settings are activated.
4		Notice	Disable	Configuration changed failed	A user changed the configuration setting, but the new settings activated failed.
5		Warning	Disable	Power input failure	The device detects power input but doesn't provide electricity (only happens on multiple power input models).
6		Informational	Disable	NTP success	The device synchronizes the time with the NTP server successfully.
7		Warning	Disable	NTP fail	The device failed to synchronize the time.
8		Informational	Disable	Manual setting time success	Manual setting time success.
9		Notice	Disable	Email fail	The device failed to deliver the email message.
10		Notice	Disable	SNMP inform fail	The device failed to deliver the SNMP Inform message.
11		Notice	Disable	Syslog fail	The device failed to deliver the Syslog message.
12		Notice	Disable	Email service is back	Email service resumed; the event recorded for successfully sending after a failure.
13		Notice	Disable	SNMP inform service is back	SNMP information service resumed; the event recorded for successfully sending after a failure.
14		Notice	Disable	Syslog service is back	Syslog service resumed; the event recorded successfully after a failure.
15		Informational	Disable	LCM display ready	The system detects the LCM display, and it's ready for use.
16		Notice	Disable	LCM display does not work	The system detects the LCM display, but it doesn't work.
17		Informational	Disable	Ethernet link up	The Ethernet port is linked up.
18		Notice	Disable	Ethernet link down	The Ethernet port is linked down.
19		Notice	Disable	IP changed	A user changed the network configuration setting, and the new settings are activated.
20	Network	Error	Disable	IP conflict	The device detects an IP conflict; this may make the device malfunction.
21		Warning	Disable	Not getting IP from the DHCP server	The device shall get an IP address from the DHCP server, but it failed.
22		Warning	Disable	Connect DHCP server fail	The device cannot find a DHCP server on the network.
23		Notice	Disable	Using 169.254.x.x IP	The device is using 169.254.x.x IP address, which is abnormal.

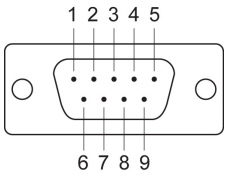
Item	Category	Severity	Default Setting	Event Name	Description
24		Informational	Disable	IP renew	IP of the device is renewed (with DHCP enabled).
25		Notice	Disable	Topology change	When Redundant protocol (RSTP or Turbo Ring) is enabled, the port on the server is blocked to prevent data loops. When the network server path is broken and communication is working with the client path (only for the models which supports Redundant protocols).

D. Pinouts and Cable Wiring

As mentioned in Chapter 2, the pin assignment of MGate MB3000-G2 Series is as below:

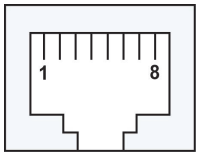
The serial port RS-232/422/485 pin assignment (male DB9):

Pin	RS-232	RS-422 4-wire RS-485	2-wire RS-485
1	DCD	TxD-(A)	–
2	RxD	TxD+(B)	–
3	TxD	RxD+(B)	Data+(B)
4	DTR	RxD-(A)	Data-(A)
5	GND	GND	GND
6	DSR	–	–
7	RTS	–	–
8	CTS	–	–
9	–	–	–



The Ethernet port pin assignment (RJ45):

Pin	RJ45
1	Tx+
2	Tx-
3	Rx+
4	–
5	–
6	Rx-
7	–
8	–

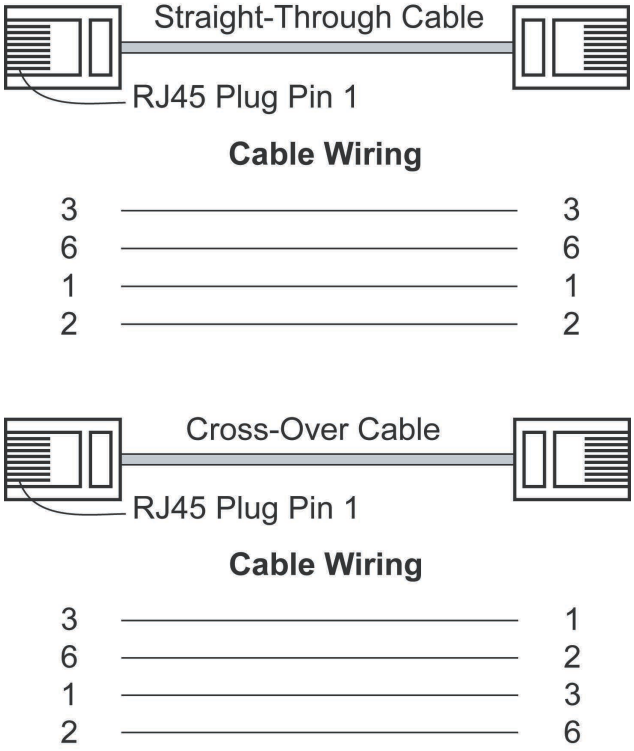


Cable Wiring Diagrams

To connect the serial devices/Ethernet devices, customize the connecting cable to connect the MGate and the serial/Ethernet devices. Here are some of the most popular cable wirings for your reference.

Ethernet Cables

There are two major types of RJ45 Ethernet cable: straight-through and crossover cables.



Serial Cables

Depending on the different connectors on the serial devices, we provide several serial cables to connect easily to the MGate and the device.

CBL-RJ45F9-150

The CBL-RJ45F9-150 is a 150-cm long cable to connect the MGate's DB9 male connector to a serial device with RJ45 serial connector. The pin assignment of this cable is as below:

Pin on DB9 male	RS-232 signal
1	DCD
2	RxD
3	TxD
4	DTR
5	GND
6	DSR
7	RTS
8	CTS
9	–

Pin on RJ45	RS-232 signal
6	DCD
4	RxD
5	TxD
1	DTR
3	GND
8	DSR
7	RTS
2	CTS
–	–

CBL-RJ45SF9-150

Industrial applications such as the factory floor are typically electrically noisy environments. The CBL-RJ45SF9-150 is a 150-cm long cable, shielded to protect the signals from the noise and connect the MGate's DB9 male connector to a serial device with a RJ45 serial connector. The pin assignment of this cable is as below:

Pin on DB9 male	RS-232 signal
1	DCD
2	RxD
3	TxD
4	DTR
5	GND
6	DSR
7	RTS
8	CTS
9	–

Pin on RJ45	RS-232 signal
6	DCD
4	RxD
5	TxD
1	DTR
3	GND
8	DSR
7	RTS
2	CTS
–	–

CN-20070

The CN-20070 is a 150-cm long cable that connects the MGate's DB9 male connector to a serial device with a 10-pin RJ45 serial connector. The pin assignment of this cable is as below:

Pin on DB9 male	RS-232 signal
1	DCD
2	RxD
3	TxD
4	DTR
5	GND
6	DSR
7	RTS
8	CTS
9	–
10	–

Pin on 10-pin RJ45	RS-232 signal
1	DCD
5	RxD
6	TxD
2	DTR
7	GND
9	DSR
8	RTS
3	CTS
–	–
–	–

E. Accessory Introduction

Moxa provides different accessories for different user scenarios. The scenarios will be introduced with the appropriate accessory in this appendix.

Convert the DB9 Connector to Other Connectors

The DB9, RJ45 and terminal block are the most popular interfaces on serial communications. The MGate has a built-in DB9 connector as the default. Moxa provides a connector to convert the DB9 interface to other connectors.

ADP-RJ458P-DB9F

The ADP-RJ458P-DB9F is a connector that transforms the MGate's DB9 male connector to an 8-pin RJ45 serial connector. The pin assignment of this connector is as below:

Pin on DB9 male	RS-232 signal
1	DCD
2	RxD
3	TxD
4	DTR
5	GND
6	DSR
7	RTS
8	CTS
9	–

Pin on RJ45	RS-232 signal
6	DCD
4	RxD
5	TxD
1	DTR
3	GND
8	DSR
7	RTS
2	CTS
–	–

Mini DB9F-to-TB

The Mini DB9F-to-TB is a connector that transforms the MGate's DB9 male connector to a 5-pin terminal block serial connector. This connector usually is used in a RS-422/RS-485 application. The pin assignment of this connector is as below:

Pin on DB9 male	RS-422 signal
1	TxD-(A)
2	TxD+(B)
3	RxD+(B)
4	RxD-(A)
5	GND
6	–
7	–
8	–
9	–

Pin on RJ45	RS-422 signal
2	TxD-(A)
1	TxD+(B)
3	RxD+(B)
4	RxD-(A)
5	GND
–	–
–	–
–	–
–	–

Pin on DB9 male	4w RS-485 signal
1	TxD-(A)
2	TxD+(B)
3	RxD+(B)
4	RxD-(A)
5	GND
6	–
7	–
8	–
9	–

Pin on RJ45	4w RS-485 signal
2	TxD-(A)
1	TxD+(B)
3	RxD+(B)
4	RxD-(A)
5	GND
–	–
–	–
–	–
–	–

Pin on DB9 male	2w RS-485 signal
1	–
2	–
3	Data+(B)
4	Data-(A)
5	GND
6	–
7	–
8	–
9	–

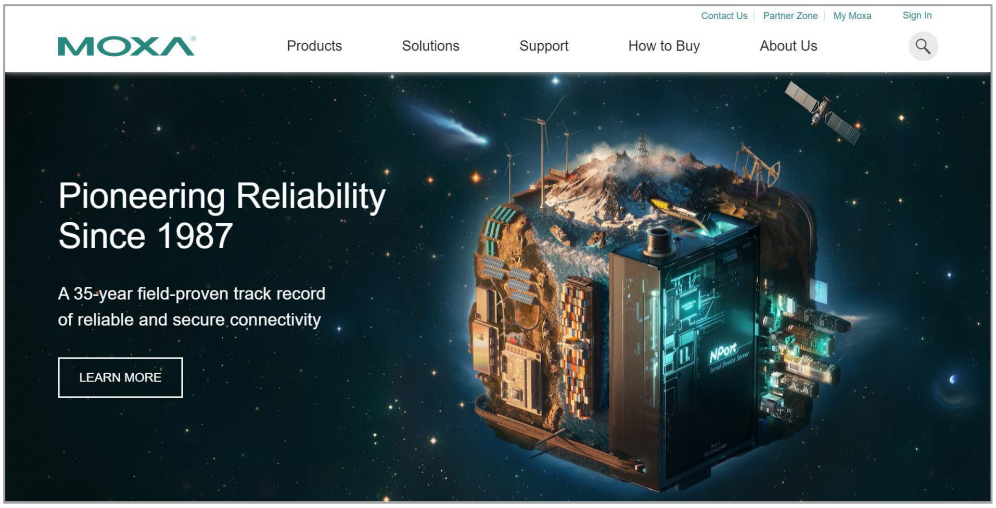
Pin on RJ45	2w RS-485 signal
–	–
–	–
3	Data+(B)
4	Data-(A)
5	GND
–	–
–	–
–	–
–	–

F. How to Become a Registered User

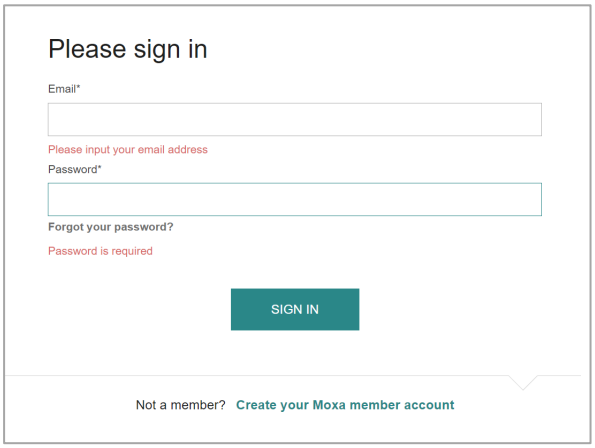
By becoming a registered user on Moxa.com, you gain access to all updates for your purchased or interested products, including software and documentation. To become a registered user and receive all updates, you need to do the following:

Register a Moxa Account

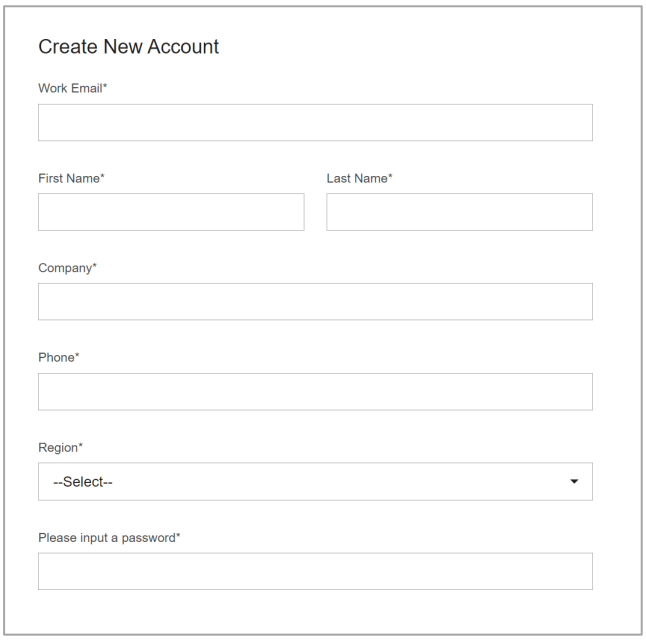
1. Go to Moxa.com and select **'Sign in'** at the top-right corner.



2. In the Sign-n page, select "Create your Moxa member account" as below.'



3. Fill the necessary fields.



Request for Product Updates

1. Go to the specific product page to receive updates. Select **"+FOLLOW UPDATE"**

2. Once completes, see the FOLLOW UPDATES button changes.