GUIDE TO NETWORKING *GROOV*

Form 2161-180108—January 2018

OPTO 22
Automation made simple.

43044 Business Park Drive • Temecula • CA 92590-3614
Phone: 800-321-OPTO (6786) or 951-695-3000
Fax: 800-832-OPTO (6786) or 951-695-2712
www.opto22.com

Product Support Services
800-TEK-OPTO (835-6786) or 951-695-3080
Fax: 951-695-3017
Email: support@opto22.com
Web: support.opto22.com

Guide to Networking *groov*
Form 2161-180108—January 2018

The information in this manual has been checked carefully and is believed to be accurate; however, Opto 22 assumes no responsibility for possible inaccuracies or omissions. Specifications are subject to change without notice.

Opto 22 warrants all of its products to be free from defects in material or workmanship for 30 months from the manufacturing date code. This warranty is limited to the original cost of the unit only and does not cover installation, labor, or any other contingent costs. Opto 22 I/O modules and solid-state relays with date codes of 1/96 or newer are guaranteed for life. This lifetime warranty excludes reed relay, SNAP serial communication modules, SNAP PID modules, and modules that contain mechanical contacts or switches. Opto 22 does not warrant any product, components, or parts not manufactured by Opto 22; for these items, the warranty from the original manufacturer applies. Refer to Opto 22 form 1042 for complete warranty information.

Wired+Wireless controllers and brains are licensed under one or more of the following patents: U.S. Patent No(s). 5282222, RE37802, 6963617; Canadian Patent No. 2064975; European Patent No. 1142245; French Patent No. 1142245; British Patent No. 1142245; Japanese Patent No. 2002535925A; German Patent No. 60011224.

Opto 22
Automation Made Simple.

Table of Contents

1: Networking Basics

INTRODUCTION

We live in an increasingly connected world. Computers, wearables, and mobile devices are proliferating, with new features and capabilities appearing in a wide variety of devices. To no one's surprise, automation engineers and technicians want to take advantage of these new abilities to monitor and control their systems, both within their company facility and remotely.

And now that many control systems are moving away from proprietary buses and into standard networks and protocols—like standard IEEE 802.3 wired Ethernet networks and IEEE 802.11 wireless networks—this kind of communication with computers and mobile devices is much easier.

About the *groov* Edge Appliance

The *groov* Edge Appliance (also called the *groov* Box) is Opto 22's internet of things (IoT) and operator interface appliance that provides visualization, data handling, and connectivity to automation systems, software, databases, and devices of all kinds—all in a compact, industrially hardened box suited to the edge of the network.

Visualization. With browser-based *groov* View, you can easily build an operator interface to see and interact with data from sensors and automation systems, cloud applications, databases, web services, and more. Authorized users can securely view your interface on any brand device, from a smartphone to a computer to a web-enabled big-screen TV. They can also receive email or text notifications for events you configure.

Data Handling. Standard Internet and IT-compatible tools are built into the *groov* Box, so you can manipulate and move data between things in the real world and computer systems and software. Tools include Node-RED, an IoT rapid application development environment; RESTful APIs to *groov* Data Stores; and Ignition Edge® from Inductive Automation®, with OPC-UA drivers and the lightweight MQTT transport protocol with Sparkplug payload. (A *groov* Enterprise license is required for OPC UA and MQTT.)

Connectivity. *groov* simplifies the connections you need to accomplish your IoT goals. Connect to all kinds of devices and systems to monitor and control them and move data between them: Modbus®/TCP devices, Allen-Bradley® and Siemens® PLC systems, Opto 22 SNAP PAC controllers, other automation equipment, cloud applications, IoT platforms, on-premises databases, and external web services.

For more information on the capabilities of the *groov* Box, see the *groov Box User's Guide* (form 2104).

What's in this Guide

Networking can be a complex subject. This guide tries to reduce the complexity by providing guidelines for how you might set up communications between your *groov* Box, your automation systems and equipment, and software or services either on premises or outside in the cloud.

The goal is for you to be able to view and control your equipment and data from anywhere you need to, either inside your facility or outside it. (If you're a machine builder or OEM, be sure to see the section "OEMs and Machine Builders" on page 20.)

This guide shows you how to communicate with *groov* using wired Ethernet networks and wireless LANs. It does not cover serial networking or other kinds of networks.

This guide includes:

Chapter 1: Networking Basics—This chapter, which introduces basic networking concepts you need to know

Chapter 2: Communication within your Facility—Setting up communications internally

Chapter 3: Communication over the Internet—Setting up remote communications using the Internet

Chapter 4: Glossary and Resources—Definitions of common networking terms as they apply to this guide, plus some resources online that may help you

For Help

For help on Ethernet networking, setting up VPNs, and port forwarding, many good resources are available online. One we recommend is Whatismyip.com, which includes FAQs on a number of subjects plus a forum for asking questions.

Related Documents and Forum

Be sure to check the user's guides for help with *groov.* All guides are available on our website at any time. Follow the links below or go to www.opto22.com and search on the form number.

Guide name	Contents	Form #
groov Box User's Guide for GROOV-AR1	Installing the *groov* Edge Appliance; using Node-RED to build logic flows; using Ignition Edge OPC-UA drivers and MQTT/Sparkplug	2104
groov Build and View User's Guide	Using groov View to build an operator interface that runs on computers and mobile devices	2027
groov Server for Windows User's Guide	Installing and using groov Server on a Windows computer	2078

The ***groov* Forum** is also available 24 hours a day, 7 days a week, so you can get advice from experienced *groov* users.

Product Support

If you can't find the help you need in this guide or in the user's guides, contact Opto 22 Product Support. Product Support is free.

Phone:	800-TEK-OPTO (800-835-6786 toll-free in the U.S. and Canada) 951-695-3080 Monday through Friday, 7 a.m. to 5 p.m. Pacific Time
Fax:	951-695-3017
Email:	support@opto22.com
Opto 22 website:	www.opto22.com

NOTE: Email messages and phone calls to Opto 22 Product Support are grouped together and answered in the order received.

CONNECTING TO COMPUTERS

How does the data get there?

NOTE: See Chapter 4: Glossary and Resources for more information about the terms used in this guide.

We all know that computers and other electronic devices—printers, routers, laptops, smartphones, and more—are networked so they can exchange information. But how does that information get where it's supposed to go? How does a spreadsheet get to the printer, a YouTube video get to your smartphone, or a value from a sensor get to your computer?

Request and response

Computers communicating on a network typically use the request-response model. (Next we'll look at a *publish-subscribe* model, which is different.) In the request-response model, a client computer or software requests data or services, and a server computer or software responds to the request by providing the data or service.

For example, when you send a spreadsheet to the printer, your spreadsheet program is the client. Its request for printer service goes to your company's print server, which responds to the request and allocates resources for printers on the network. The print server handles all the client requests for printing, making sure your spreadsheet and your coworkers' print jobs are all completed in an orderly way.

When you want to watch that YouTube video on your smartphone, your web browser or YouTube app is the client, requesting the video over that giant of networks, the Internet. YouTube's web server receives the request and responds by serving the video page to you, along with the other millions of video pages going to other millions of viewers worldwide.

groov is a server and responds to requests

The *groov* Box and *groov* Server for Windows both act as web servers for your *groov* View operator interface. At the request of clients like authorized smartphones and tablets, *groov* responds by serving the interface pages you've created to these clients on the network.

Node-RED nodes are clients and send requests

Within the *groov* Box, the nodes you use in your Node-RED flows are clients. They send requests to servers to request resources (like data from an online weather service) or services (like pushing data to a database).

Ignition Edge is a server that responds to internal requests only

Within the *groov* Box, Ignition Edge provides an internal OPC-UA server and drivers (a *groov* Enterprise license is required to use Ignition Edge). *groov* View and Node-RED are the only clients for the Ignition Edge OPC-UA server; they send requests to it from within the Box using a local address. No clients outside the Box can access this internal OPC-UA server.

Publish and subscribe

A different way for devices to communicate on a network is called publish-subscribe, or *pub-sub*. In a pub-sub architecture, a central source called a broker (also sometimes called a server) receives and distributes all data. Pub-sub clients can publish data to the broker or subscribe to get data from it—or both.

Clients that publish data send it only when the data changes. Clients that subscribe to data automatically receive it from the broker/server, but again, only when it changes.

The broker does not store data; it simply moves it from publishers to subscribers. When data comes in from a publisher, the broker promptly sends it off to any client subscribed to that data. You can think of data from a publisher as an incoming shipment on a truck. The broker sees the truck come in but doesn't unload it; it simply routes it intact to a subscriber (cloning the truck if there's more than one subscriber).

MQTT—a pub-sub protocol

Message Queue Telemetry Transport (MQTT) is a fairly well-known transport protocol that uses the pub-sub architecture. MQTT is extremely lightweight: it takes up almost no space in a device, so that even small devices with very little computing power can use it.

In our analogy, MQTT defines the truck and the routes. But it doesn't define how the load (the data) is packed or unpacked. That's where Sparkplug comes in.

MQTT with Sparkplug

The Sparkplug open MQTT client specification provides a messaging format appropriate for industrial use. Sparkplug encodes the data payload: it defines how the data is packed on the truck before it's sent by the publisher, and how it is unpacked in the subscriber.

Data sent over MQTT with Sparkplug is compressed and efficient. MQTT trucks that have been packed with the Sparkplug definition must also be unpacked with Sparkplug, so both publishers and subscribers must use it in order to get the data delivered.

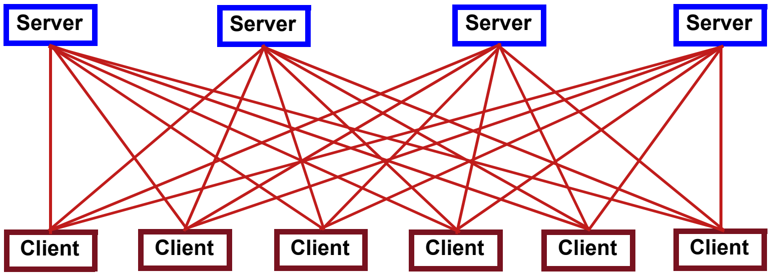
MQTT with Sparkplug also provides an efficient way to track the state of clients and make sure that clients on a tenuous connection can still deliver or receive data. If the client goes offline (breaks its connection with the broker), the broker sends a "death certificate" to clients subscribed to that data. When the client comes back online (re-establishes the connection), the broker issues a "birth certificate" with the current status of all data tags. A certain amount of missed data can also be sent, depending on client configuration.

Comparison: request-response and pub-sub

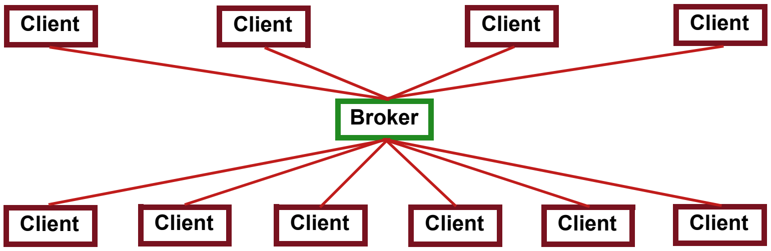
In a request-response architecture, each client must open a direct connection to each server, because the client requests data directly from the server. Also, because the client doesn't know when data may change, it must request it at regular intervals. So clients are repetitively sending requests to servers—often once per millisecond—and servers are repetitively responding:

Q: What's the sensor value? A: 10
Q: What's the sensor value? A: 10
Q: What's the sensor value? A: 10
Q: What's the sensor value? A: 10
Q: What's the sensor value? A: 10
Q: What's the sensor value? A: 9
Q: What's the sensor value? A: 9

If you have multiple servers and multiple clients, the volume of traffic can quickly become a problem. Below you see an example of the request-response model. Each client is individually connected to each server it needs to request data from, and each connection may even be opened, queried, answered, and shut, over and over:



In contrast, a pub-sub architecture simplifies communications. Direct connections and repetitive requests for data are not needed. The web of links is replaced by a single link from each device to the broker. The connection between client and broker is kept open and is incredibly lightweight. Only two things travel over this connection: changed data, and a tiny heartbeat to let the broker know that the client is still there.



MQTT/Sparkplug in the *groov* Edge Appliance

Ignition Edge in the *groov* Box offers MQTT transport with Sparkplug messaging. To use it you'll need a *groov* Enterprise license, but you can try it on a repeatable two-hour trial for free.

You will also need an MQTT broker, which you can locate on your premises or in the cloud. For a list of public brokers useful for prototyping, see https://github.com/mqtt/mqtt.github.io/wiki/public_brokers. For more on the protocol, see mqtt.org. And for more on using Ignition Edge and MQTT/Sparkplug, see the *groov Box User's Guide*.

For industrial internet of things (IIoT) applications, MQTT/Sparkplug can offer several key advantages over request-response:

- Since the broker is the central clearinghouse for data, individual servers don't have to strain to serve multiple clients, and clients don't have to connect to multiple servers.
- Network traffic is reduced overall, because data is published and sent only when it changes, rather than at regular intervals.
- Because payloads are compressed and data moves efficiently, even remote devices with irregular connections or low bandwidth can publish or subscribe to data.
- Devices that go offline can reconnect with the broker, sending or receiving current data and also a specified amount of buffered data to help fill in the gap.
- For data publishers, there's another important advantage: data is published using an outbound connection. All firewalls block inbound traffic (for example, an external client requesting data from an internal server), but they typically allow outbound connections over TCP ports.

This last point is a key consideration for setting up networking. Because data is sent from devices and software using only outgoing communications (to the broker), these communications do not require either a VPN or port forwarding.

IP addresses

How does a client reach a server or a broker? It's similar to the way you call someone on your cell phone. You tap their name, the phone dials their phone number, and the phone system understands how to connect to the phone at that number. The format of the phone number tells the system how to connect.

In computer networking, the equivalent of a phone number is an IP address. Most of us don't have to pay attention to IP addresses, just like we don't memorize our friends' phone numbers. It's harder to remember a long number than a name (and computer IP addresses can change). So instead of typing the IP address, we click a printer name or enter a domain name like youtube.com or opto22.com.

But in the background, computer networks, just like the phone system, know how to make the connection. A domain name server (DNS) translates the device name or domain name into an IP address. Routing tables and software rules tell routers how to send your packets of data to the right destination.

Sometimes a computer network is very small—so small that both client and server in a request-response architecture are on the same computing device. For example:

- When you load *groov* Server for Windows on your PC, you access *groov* View from the same computer by using the name `localhost` or the equivalent IP address: `127.0.0.1`
- When your *groov* View operator interface and your Node-RED project on your *groov* Box get data from the internal Ignition Edge OPC-UA server, they access the server by using the localhost address and port.

NETWORKING WITHIN YOUR FACILITY

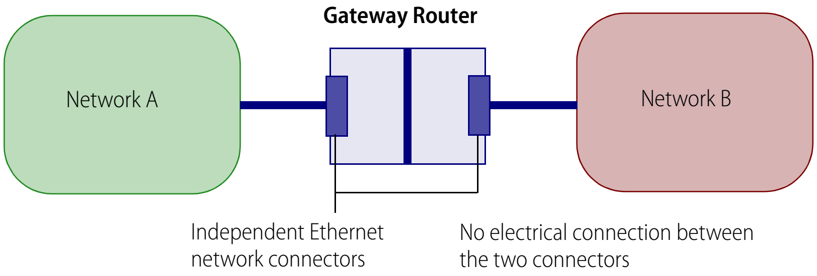
Within your facility you may have one or more subnetworks or local area networks (LANs).

Maybe you have all your devices on a single flat network: your computers, printers, wireless access points, and control system are all on one LAN, so all these devices can freely communicate. This network architecture makes communication simple (see "Single flat network" on page 9).

But many companies have more than one LAN. You may have your control system on a separate network from your company computers, for example, to keep the control system segmented for less traffic and increased security. If you want a person or device on one LAN to communicate with a person or device on another, you need a *gateway router*.

How a gateway router works

A gateway router is wired to both subnetworks through independent Ethernet network interfaces. Communication between these two interfaces can occur only if the software rules inside the router allow it. These software rules typically include routing tables and network address translation (NAT).


Gateway Router
Network A
Network B
Independent Ethernet network connectors
No electrical connection between the two connectors

Software rules (routing tables, network address translation) determine whether and how communication moves between Network A and Network B.

In addition to managing communication between LANs within your facility, a gateway router is also used to manage communication between a LAN and a WAN (wide area network). A WAN may be private or public; the Internet is a public WAN.

The gateway router acts in exactly the same way whether it's managing communication between two LANs or between a LAN and a WAN. The LAN is plugged into one Ethernet network interface on the router and the WAN is plugged into another. Communication occurs only as allowed by software inside the router.

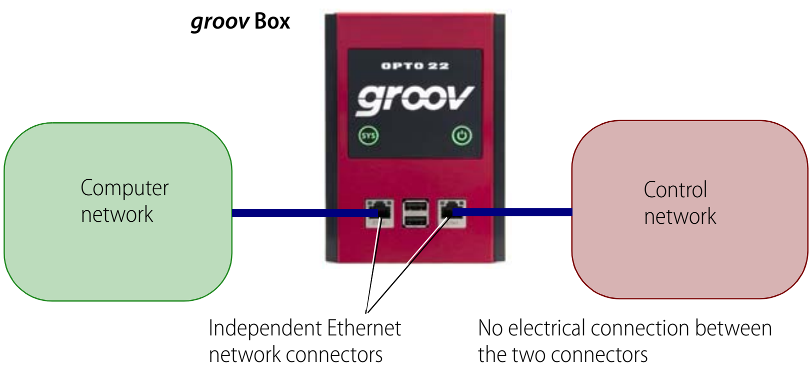
We'll talk more about networking over the Internet in Chapter 3.

The *groov* Box

Like a gateway router, the *groov* Box has two independent Ethernet network interfaces (and in some cases also an independent wireless interface). Each of these independent interfaces, if used, must be wired to a separate network. That means their network addresses (a combination of IP addresses and subnet masks) must be different.

groov Boxes are not routers, because they do not provide routing or address translation, but their separate interfaces work like a router's interfaces. If your control network is wired to ETH0 on the *groov* Box and your computer network is wired to ETH1, the two networks are segmented. Data packets cannot travel between them.

So if you've built a *groov* View interface with a switch to turn on a pump, an authorized user can switch on the pump. But he cannot control a valve that isn't in the interface or that's on a screen he's not authorized to see. Nor can he directly access any systems on the other network.



***groov* Box**
OPTO 22
groov
SYS
Computer network
Control network
Independent Ethernet network connectors
No electrical connection between the two connectors

Your *groov* project software determines what data a *groov* View user can see and change.

Remember: The independent network interfaces on *groov* Boxes must always be on separate networks (using different IP addresses and subnet masks). *groov* Boxes do not route IP traffic; they have no routing tables or network address translation. There is no communication between the two networks.

HOW DO YOU WANT TO USE *groov*?

In the next two chapters we'll take a look at your network setup and how you may want to use the *groov* Edge Appliance in it:

- Use *groov* in your facility with one Ethernet network—page 9
- Use *groov* View for mobile in your facility: easy setup for a wireless network (SoftAP)—page 10
- Use *groov* in your facility with segmented systems—page 12
- Solve facility network issues with pub-sub—page 14
- *groov* for OEMs and machine builders—page 20
- Beyond your facility: communicating over the Internet—page 15
- Networking options: MQTT/Sparkplug, VPN, port forwarding—page 18

2: Communication within your Facility

INTRODUCTION

Inside your facility, you may want to have computers and/or mobile devices communicate with your control system. Maybe you want to monitor production numbers, send equipment data to a database or spreadsheet, operate machinery, or control processes. How you do so depends on your network setup:

- Everything is on one network. See "Single flat network," below.
- Two or more networks exist—for example, a company computer network and a control system network. See "Two or More Networks" on page 10.

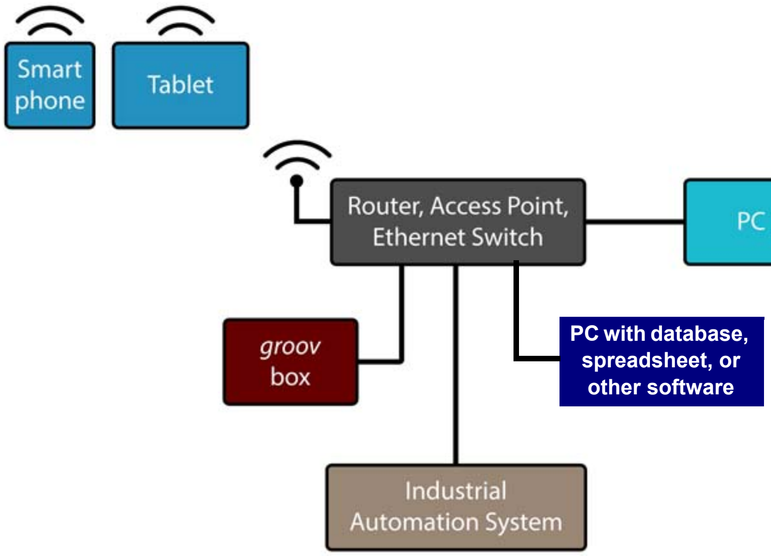
SINGLE FLAT NETWORK

Use *groov* in your facility with an existing Ethernet network

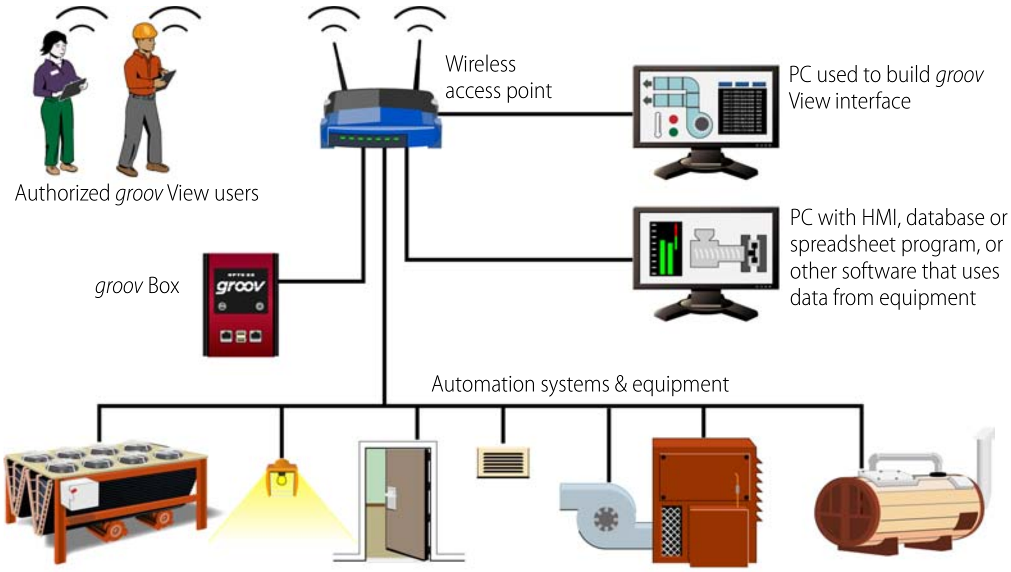
A single network simplifies setup. If you want to use *groov* within your facility only (not remotely) and you already have a wired Ethernet network in place for your automation system, you can just plug *groov* in. Authorized users (human or software) can use data, or monitor and control equipment with your *groov* View operator interface, as long as they are on the wired Ethernet network. If you have users on mobile devices, you need a wireless network, too.

The following diagrams show the basic architectural components:

- You plug the *groov* Box into the same Ethernet network as the industrial automation system.
- *groov* View can connect to Opto 22 SNAP PACs and Modbus/TCP devices on the network, while the Ignition Edge OPC-UA drivers in the *groov* Box can connect to PLCs and other equipment.
- Both *groov* View and Node-RED in the *groov* Box can connect to PCs on the network running database and spreadsheet software, to show or store data.
- You build your *groov* View operator interface on a PC that is on the same wired network as *groov*.
- Authorized users can access data and use your interface from computers on the same wired network.
- With a WiFi connection, authorized users can also use your operator interface from their mobile devices.


Smart phone
Tablet
Router, Access Point, Ethernet Switch
PC
groov box
PC with database, spreadsheet, or other software
Industrial Automation System

Here's the same network, but with people and systems in place of boxes:


Wireless access point
PC used to build *groov* View interface
Authorized *groov* View users
PC with HMI, database or spreadsheet program, or other software that uses data from equipment
groov Box
Automation systems & equipment

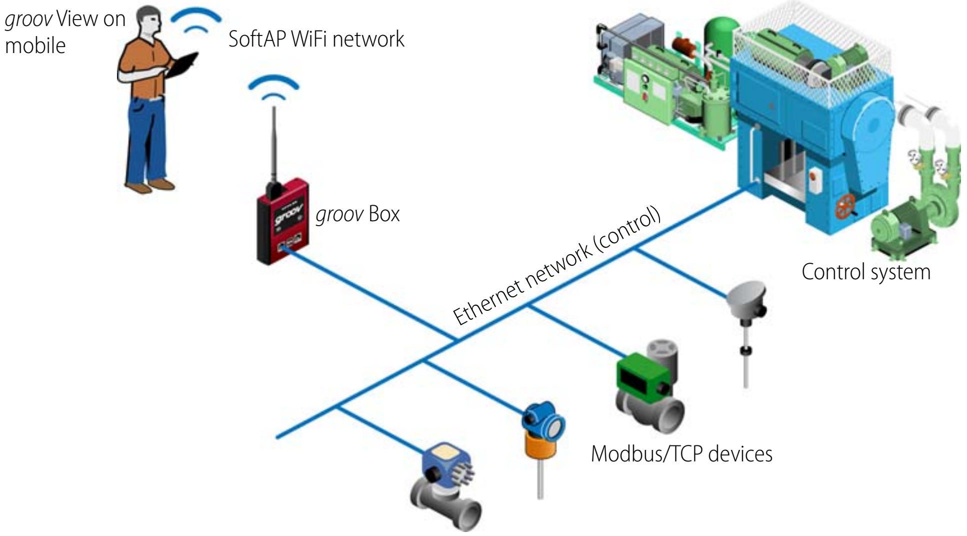
TWO OR MORE NETWORKS

Add a wireless network for mobile users

If a wireless network does not exist in your facility, but mobile device users need to connect wirelessly to *groov,* you can configure a *groov* Box to create its own private WiFi network with WPA2-PSK security. This feature is called **SoftAP** (software enabled wireless access point).

To set up SoftAP, you'll need a GROOV-AR1 with groov Admin v 1.570.39 or higher, and an approved wireless adapter. See the *groov Box User's Guide* (form 2104, Chapter 3 and Appendix D) for a list of compatible adapters, plus details on configuring SoftAP and installing the adapter.

The following image shows two network subnets separated by a *groov* Box. The wired Ethernet network is for control. The SoftAP wireless network is for nearby mobile devices using *groov* View; these can include any WiFi-capable device, such as phones, tablets, and laptops.



Again, the mobile devices on the SoftAP WiFi network have no direct connection with the automation equipment on the control network. Authorized employees on mobile devices who join the SoftAP network can see only the data and controls you've built into the *groov* interface.

A similar network architecture can exist on a much smaller scale; for example, a machine builder or OEM can use a *groov* Box with SoftAP to provide local access to machine data and controls through the *groov* mobile app, with no impact on existing IT networks in customer facilities.

To maximize security, **SoftAP cannot be used as a wireless hotspot** and does not allow tethering. That means other devices cannot use it to connect to the Internet. SoftAP works only as an access point for local mobile devices to connect to the *groov* Box.

Notes for mobile communication with a *groov* operator interface

If you're using a smartphone or tablet on your local network to connect with a *groov* operator interface, you may need to be more specific with the URL to direct the mobile device's browser:

- On an iOS device, the browser always tries port 80 first, so the secure connection to your *groov* Box or *groov* Server may time out. To prevent this, add a colon and the port number to your *groov* hostname. For example, if
  `https://`*`hostname`* times out, try adding the port number (default is 443):
  `https://`*`hostname`*`:443` (substituting your *groov* Box's actual hostname)
- For Android, add a period and your local domain name. For example, if
  `https://`*`hostname`* results in an error, try:
  `https://`*`hostname.domainname.com`* (substituting the actual hostname of your *groov* and your company's domain name)

With two or more wired networks within your facility, setup becomes a little more complex. But there are still ways to move and view data securely and efficiently.

Use *groov* in a facility with segmented systems

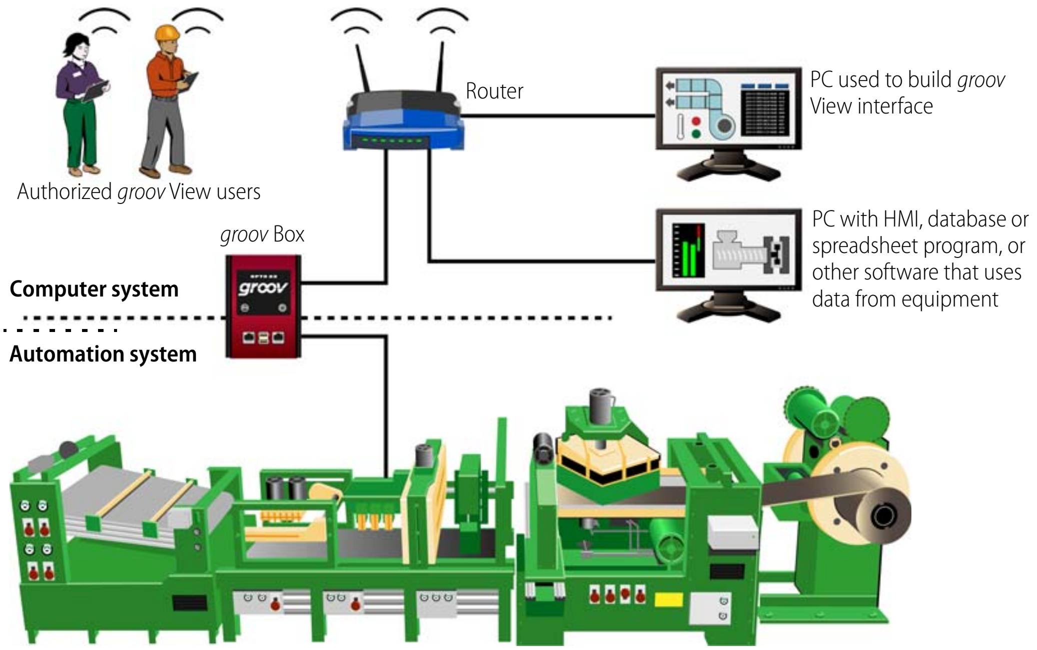
For security reasons, you may choose to take advantage of the multiple network interfaces on your *groov* Box to separate your control network traffic from your computer network. In fact, we strongly recommend that architecture for *groov*. Your computer system typically has Internet access; your automation system is safer if it does not have Internet access.

As explained on page 6, the two wired Ethernet interfaces (and the WLAN interface, if present) on a *groov* Box are independent from each other. Data packets can't travel directly between the interfaces. The only communication that can occur between the two interfaces (and therefore between the two networks) is communication specifically allowed by the software in the *groov* Box. So authorized users can see the *groov* interface and use data, but users do not have access to the system itself.

If you're using the *groov* Box to segment networks, then you have separate network subnets.

IMPORTANT: *Because their network interfaces are not connected, you must ALWAYS assign the network interfaces on a groov Box* ***different IP addresses and different subnets****. For more information, see the groov Box User's Guide. Note that it doesn't matter which interface you use for the control network or the computer network, except initially when you must use the lower-numbered interface (ETH 0 on the groov Box).*

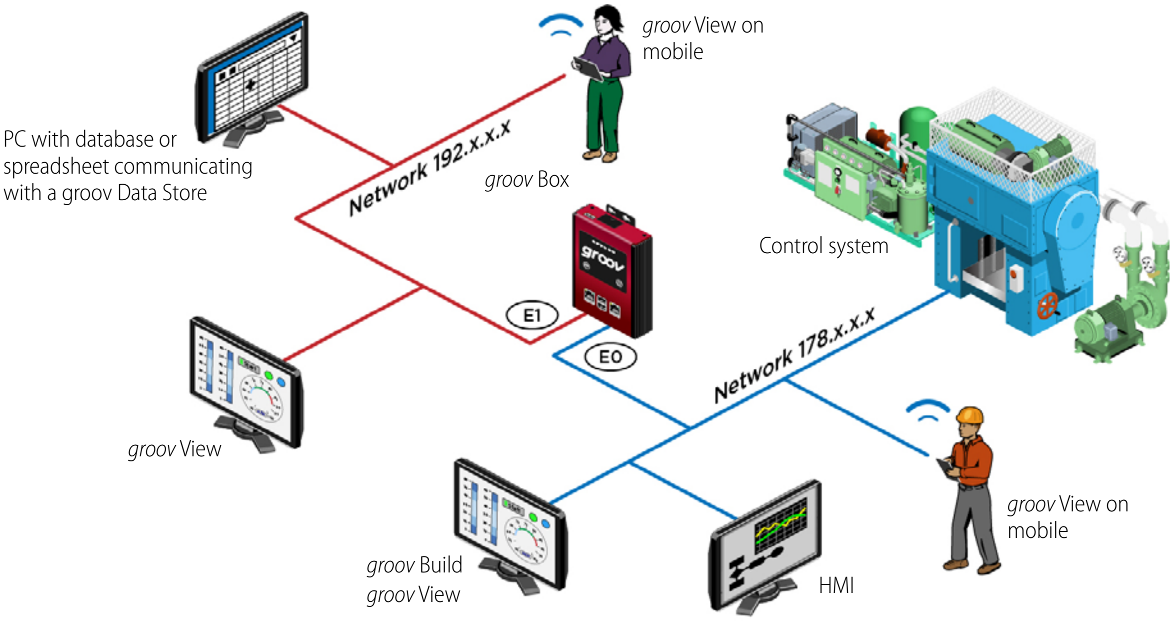
The image below shows the *groov* Box with the computer system connected to one network interface and the automation system connected to the other.


Router
PC used to build groov View interface
Authorized groov View users
PC with HMI, database or spreadsheet program, or other software that uses data from equipment
groov Box
Computer system
Automation system

Let's look at an example of separate network subnets. The key advantage to separating networks is security. Only authorized people and software can see and use the data in your *groov* View operator interface, your Node-RED project, and Ignition Edge OPC-UA tags or MQTT data in the *groov* Box. And even these authorized users can use data only in the ways you allow.

Separate network subnets with *groov*

Two network subnets offer a secure way to access specific data in specific ways. The following image shows network subnets separated by a *groov* Box. The 172.x.x.x network is for control. The 192.x.x.x network is for company computers not directly involved in the control network.



In this example, your *groov* Box is connected to your company computer network using ETH1 and to your control network using ETH0. (You could do the same thing with *groov* Server for Windows running on a PC that has two network interface cards.)

The PCs and mobile devices on the computer network have no direct connection to the equipment on the control network, so employees on these devices cannot access it directly. But your *groov* Box software can serve authorized users:

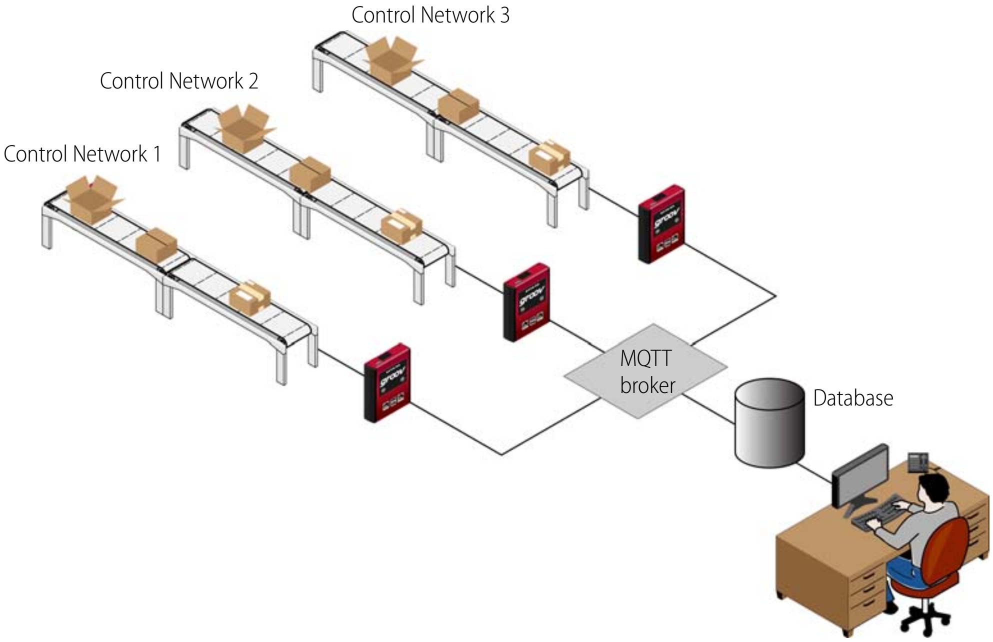
- With your *groov* View operator interface, employees can get the data they need to monitor and can change the systems and equipment they are authorized to control. For example, you might give a supervisor the ability to check production figures but not turn a conveyor on or off.
- Software users like databases and spreadsheets can also get and change data in a *groov* Data Store, based on the permissions you've given them. And your Node-RED flows can obtain and move data, too.

Solve facility network issues with pub-sub

Sometimes a facility has multiple networks, for example, separate networks for controlling each production line plus a separate network for office computers. Connecting these networks can be problematic, involving IT expense, time, and security concerns.

One way to lessen these problems and still get data from production lines is to connect networks using a pub-sub communication method instead of a request-response one. MQTT/Sparkplug in the *groov* Box simplifies this solution.

Instead of setting up firewalls and VPNs for each network, place a *groov* Box on each network as shown in the following diagram, and enable MQTT/Sparkplug. For this solution, you'll need to set up a local MQTT broker and have a *groov* Enterprise license for each *groov* Box.



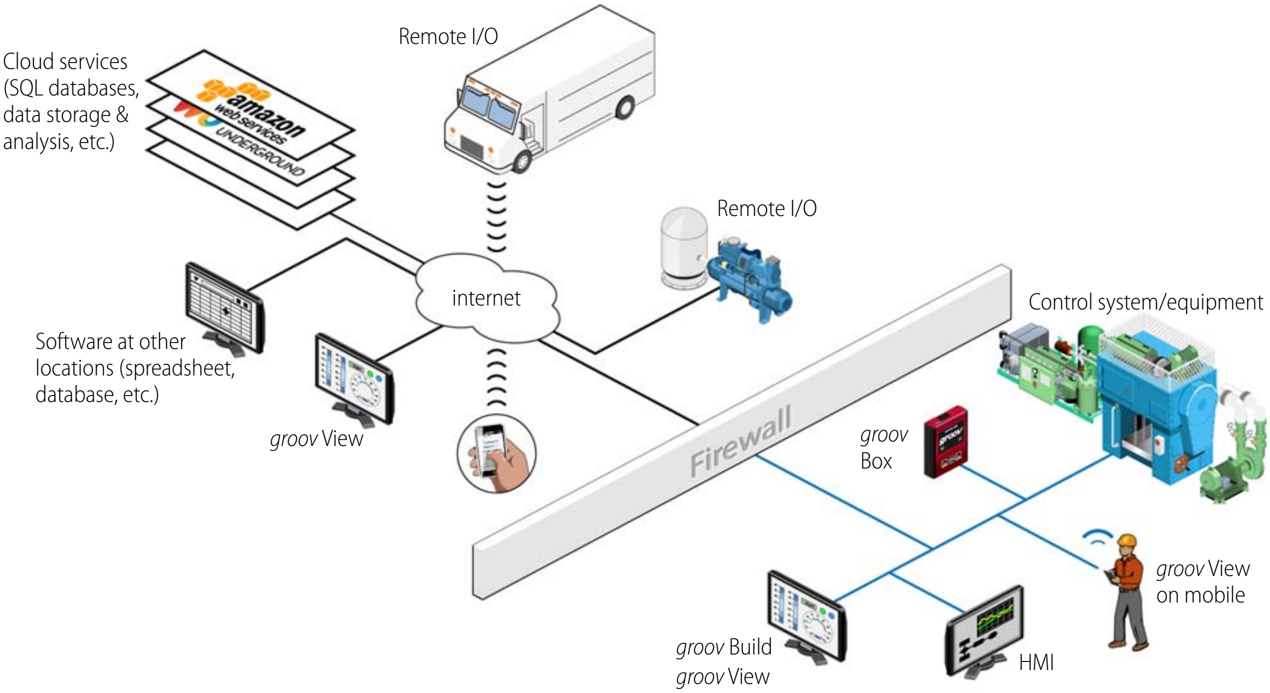
3: Communication over the Internet

BEYOND YOUR FACILITY: WHY COMMUNICATE OVER THE INTERNET?

When your control system and your company computers or mobile devices are connected by a local network, communication between them is easy. But you may have very good reasons to communicate with your control system from a different network, miles away. Here are just a few:

- An engineer needs to adjust a setpoint at another site.
- Status data from remote equipment needs to be tracked and analyzed for predictive maintenance.
- Logistics personnel need to track physical locations and other data for delivery trucks.
- Production managers want to know the number of widgets produced in the last hour, even while they're traveling.
- A technician has been notified of a malfunction in another building and needs to quickly switch from pump #1 to pump #2.

These are simple examples of the Industrial internet of things (IIoT) at work. When two or more networks are connected to the internet, devices on them can communicate. Here are some examples:



When one of the *groov* Box's interfaces can access the internet, you gain additional capabilities:

- Your Node-RED logic flows can incorporate data from a wide assortment of online services, including environmental and geographical services, regulatory information, data storage and analysis systems such as Amazon Web Services and IBM Bluemix, and much more.
- The Ignition Edge OPC drivers in the *groov* Box can tap into data from automation equipment located outside your local network and at remote locations.
- Your *groov* View operator interface can include data and controls for all these outside sources.

Any two networks can be used for the IIoT as long as both are connected to the internet:

- A computer in one location can get data from a control system at another location.
- Online software and services can supply data or receive data.
- A *groov* View operator interface on a computer or mobile device far away from your control system can access it for monitoring or control.
- A mobile device with cellular service (which goes through the internet) can use the cellular network if it can't reach the wireless LAN.

For cases like these, you can establish communication over the internet by following a few extra steps. The rest of this chapter shows you how.

CAUTIONS: SECURITY, SPEED, AND RELIABILITY

Especially in the case of sensitive data or equipment control, security is a key consideration when you're using the internet for communications. This chapter emphasizes ways to communicate in order to maximize security.

Communication speed can vary a great deal depending on your internet connection speed, the quality of the internet service provider (ISP), and even the time of day. You'll need to take this possible delay into account if you are controlling equipment or transferring data between devices.

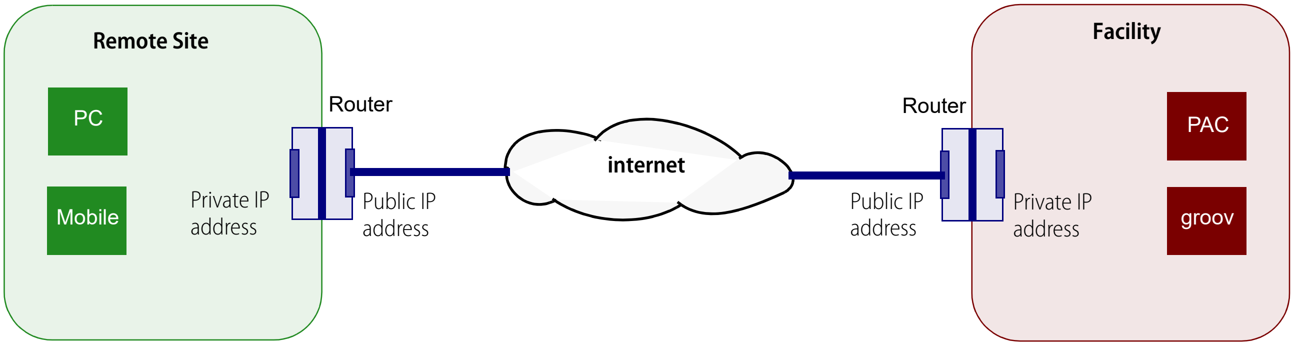
Also, because many companies and steps along the way are outside your control, you should consider the connection tenuous and plan other ways to accomplish what you need to do, in case the link goes down for a short while or for a long time.

INTERNET GATEWAY ROUTERS

Remember our gateway routers from Chapter 1 ("How a gateway router works" on page 6)? Gateway routers are essential parts of remote networking over the internet for the same reason they're essential for connecting networks within your facility: they provide security.

That's because the gateway router has two separate network interface cards (NICs), one connected to the public internet and one connected to your facility's private network. The only data that can cross to the other side is what's allowed by software rules within the router. The router's private IP address—and the IP addresses of all devices on the private network—are hidden from its public IP address.

You can see how this works in the diagram below.



When you're looking at IP addresses, the following IP addresses are always on private networks:

- 10.x.x.x
- 172.x.x.x
- 192.x.x.x

All other IP addresses are on public networks.

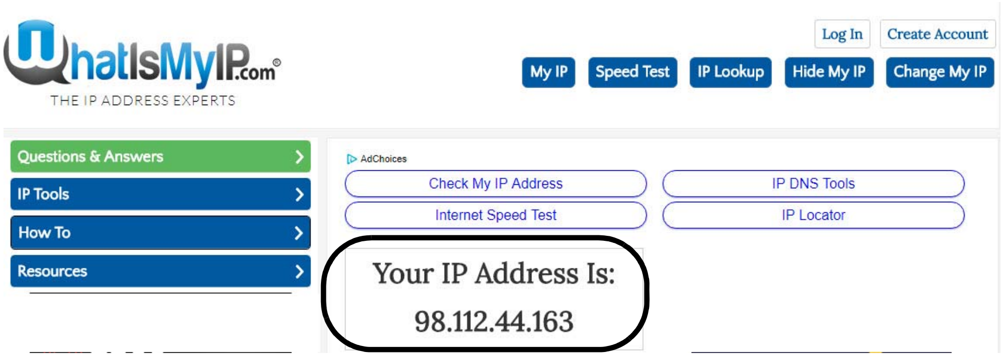
This distinction between the public and private IP addresses on the router becomes important as you set up communication.

Gateway router identification

At some point in configuring communication over the internet, you may need to know a gateway router's public IP address (also called its WAN IP address). Your internet service provider (ISP) provides this address, and the address may be fixed (static) or dynamically assigned.

1. Go to a computer that has internet access on the network whose public IP you need to know. Open a web browser and go to one of these:

   ```
   http://whatismyip.com/
   http://www.ipchicken.com/
   http://icanhazip.com
   ```



Your IP Address Is:
98.112.44.163

2. Find the IP address assigned to your company by your ISP, near the top of the page. Copy the address down exactly.

   Note that this address does not start with 10, 192, or 172. It's a public address.

Fixed (static) vs. dynamic IP addresses

As we said, the public IP address you discover may be fixed (static) and never change, or it may be dynamic and change from time to time. If you don't know, ask your ISP. (Generally you will know if it is static, because you have to pay more for a static address.)

- If the router has a static public IP address, you can use that address when setting up a VPN server or port forwarding.
- If the router has a dynamic public IP address, use a DDNS (dynamic domain name service) to assign the router a public domain name. (Remember that a DNS resolves static IP addresses into domain names; a DDNS updates DNS if your dynamic IP addresses change.)

  If your router includes a DDNS feature, set it up there. If not, set up a DDNS service on the web, for example at dyn.com/dns or noip.com. First you'll create an account on the service, and then you'll pick your domain name. Some of these services are free. Free services usually check for a change in IP address every 10 minutes. That means you might have to wait up to 10 minutes to gain remote access. You can also pay for the service and reduce the length of time between checks.

CONSIDER YOUR OPTIONS

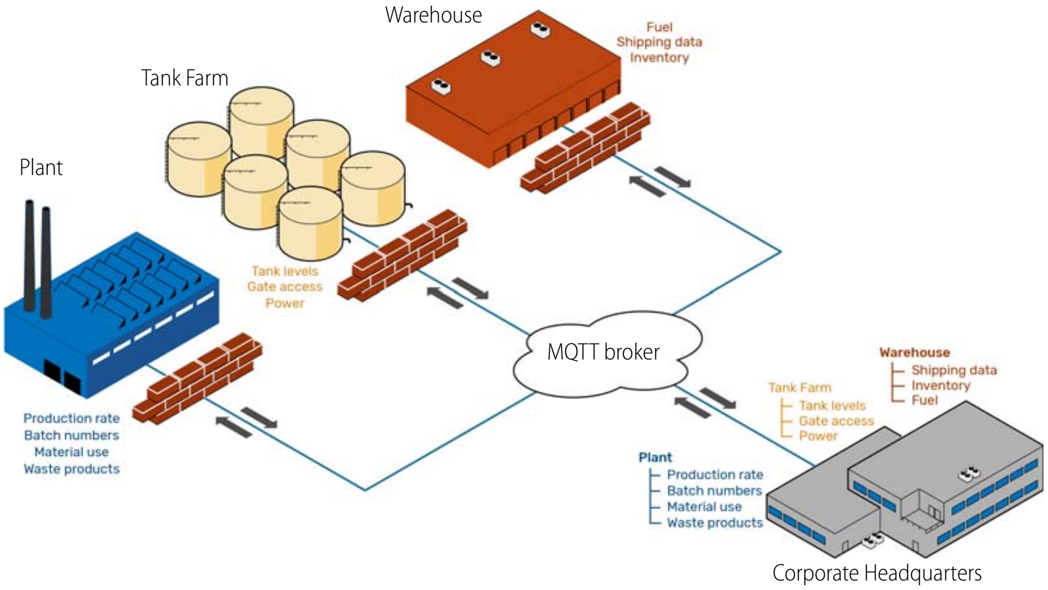
Gateway routers prevent direct communication from the internet to a private network. If you want to directly communicate with a device (like a *groov* Box) or a service (like *groov* Server) that's on a private network, you need a way around this block. You have three choices: MQTT/Sparkplug, a VPN, or port forwarding.

MQTT/Sparkplug

In many cases an MQTT/Sparkplug pub-sub architecture may suit your needs. As we discussed in Chapter 1, connections between an MQTT client and broker are always outgoing from the client. Almost all firewalls allow outgoing communications over TCP ports.

If you use the Ignition Edge MQTT/Sparkplug module in the *groov* Box for exchanging IIoT data, you will have to set up a broker, either on premises or in the cloud. But you will not have to modify your firewalls.

Here's an example of MQTT/Sparkplug used to connect remote sites and bring their data into corporate headquarters. Data communications are efficient, easier to set up, and more secure.



VPN or port forwarding

However, some actions can't be done over pub-sub. They require a request-response connection:

- Monitoring or controlling systems through a *groov* View interface from outside your facility
- Using a *groov* View operator interface on premises but out of range of a WiFi access point (because your phone switches to cellular service)
- Accessing *groov* Admin, Node-RED Admin, or Ignition Edge from outside the network the *groov* Box is on

For a request-response connection, you have two possible methods for communication over the internet: a virtual private network (VPN) or port forwarding (PF). Of these two, a VPN is preferred because it is much more secure. If you're controlling industrial equipment through your *groov* View interface or getting into *groov* Admin, you want your connection to be secure.

Here again, for security reasons we advise that you segment your computer system (which is on the internet) from your automation system (which should not be on the internet). Keeping your control system separate is one important key to keeping unauthorized people out of your systems. The *groov* Box can simplify segmentation with its two independent Ethernet network interfaces.

VPN vs. PF

A virtual private network (VPN) employs dedicated connections, authentication, and encryption to connect you to your private network from the internet, while maintaining all the same functionality and security you would have inside the network. Authentication is built into the VPN server. When you use a VPN, it's like having your own private tunnel through the internet. It feels a lot like being on site.

Port forwarding (PF) may be easier to set up than a VPN but is less secure. PF allows remote computers or mobile devices to connect to a specific computer or service within a private local area network through a specific port. Essentially it pokes a "pinhole" in your company firewall that packets of information can pass through.

In addition to security, if you have more than one *groov*, a VPN may be the better choice for practical reasons: you can set up all *groovs* at once instead of one at a time with PF.

NOTE: If you're using cellular data radio (for example, a mobile hotspot) at a remote location, check your plan for details. Some plans don't allow incoming connections to your gateway router and unfortunately won't work for either method.

If you have an IT department, work with them to set up communication over a VPN (see "Working with your IT department," below).

If you don't have an IT group, you'll have to set it up yourself. See "Setting up a virtual private network (VPN)" on page 22 or "Using port forwarding (PF)" on page 24.

WORKING WITH YOUR IT DEPARTMENT

If you have an IT department, work with them. For pub-sub, they can help you set up an MQTT broker. For request-response, they can set up the VPN, create VPN accounts for you and any other authorized users, and make sure those accounts have access to the network your *groov* is on.

The information in this guide should give you enough basic knowledge to be able to talk with your IT department about what you need. If you (or they) need more help, contact Product Support (see "For Help" on page 2).

Tell your IT department which devices you need to have communicate with each other (computer with the *groov* Box, mobile devices with *groov*) and give them a copy of this section. Then follow their instructions to set up communication on your computers and mobile devices. (For help, see "Setting up VPN clients" on page 22.

Common communications and required ports

All communications are encrypted (use https).

Communication between	Communication methods	
	VPN	PF
PC <--> *groov* Box	Yes	Yes. See ports below.
Mobile <--> *groov* View	Yes	Yes. Use port 443/8443 (TCP)

NOTE: Port forwarding for communication between groov and an Opto 22 SNAP PAC controller over a public network is NOT recommended, because the controller does not provide user authentication and encryption. For more information on networking SNAP PAC controllers, see form #1796, the Guide to Networking Opto 22 Products.

Opto 22 port usage

If your IT department plans to use port forwarding, here is the port information they need.

By default, *groov* uses the following ports for communication.

Port	Used for	Authenticated?
443 (or 8443)	*groov* View and *groov* Build	Yes
10000	*groov* Admin (applies to *groov* Box appliance only)	Yes
8043	Ignition Edge (GROOV-AR1 *groov* Box only)	Yes
3000	Node-RED Admin (GROOV-AR1 *groov* Box only)	Yes
1880	Node-RED Editor (GROOV-AR1 *groov* Box only)	Yes

OEMS AND MACHINE BUILDERS

Original equipment manufacturers (OEMs) and machine builders may find the *groov* Edge Appliance especially useful:

- For an inexpensive, off-the-shelf HMI for machine operators
- For troubleshooting and updating machines themselves, either onsite or remotely, through an operator interface
- For tracking data about the machine, through a database, spreadsheet, online service, or other software

The fanless, small-footprint *groov* Box takes up little space in your machine.

NOTE: You can do many of the same things with groov Server for Windows, if your machine already includes a PC. groov Server for Windows gives you the ability to build and view your own operator interface to use locally or remotely. However, it does not include Node-RED or Ignition Edge, so you would need to install them yourself and provide your own security for communications.

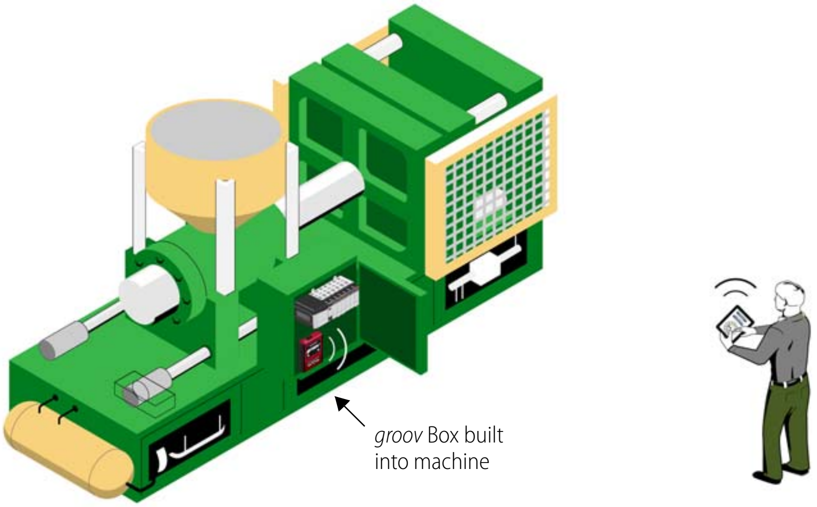
Provide a machine HMI

For visualization, you can do one or all of the following:

- Run a *groov* View operator interface on a PC connected to the same network as the machine.
- Build an off-the-shelf mobile device into the machine to use as an operator interface. On an iPhone or iPad, use Guided Access mode to lock down the device so all it does is show your *groov* interface.
- Provide a localized wireless network around the Box so operators can use it from phones, tablets, or laptops, and you or a technician can troubleshoot or update the machine onsite. To do so, use a *groov* Box with an approved WiFi adapter and SoftAP (see "Add a wireless network for mobile users" on page 10).

Remote troubleshooting using a *groov* operator interface requires that the *groov* Box in the machine be on a network with internet connectivity. For better security, connect one *groov* Box interface to the machine itself and the other to the internet network. If you go this route, we strongly recommend using a VPN (virtual private network; see page 22).

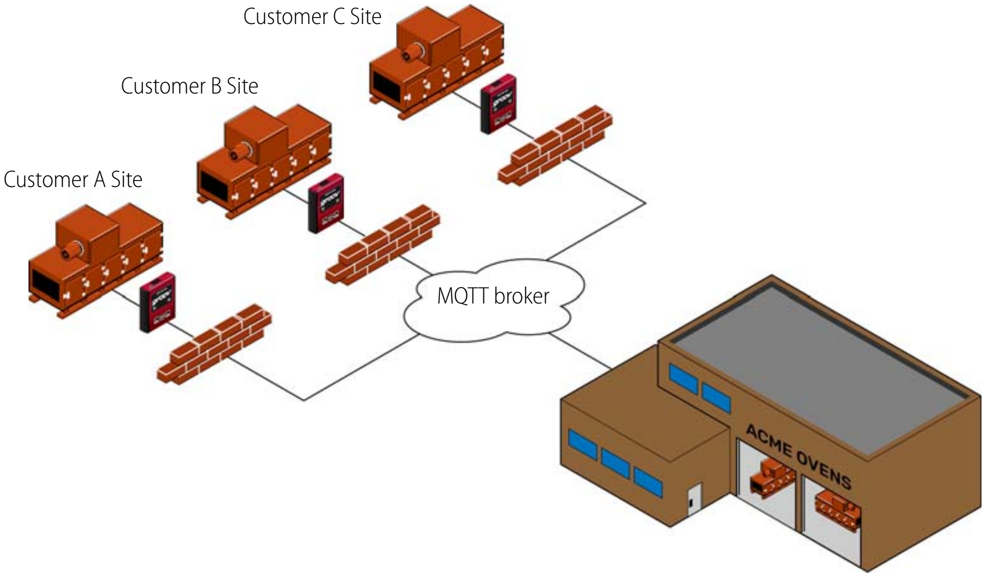
This diagram shows the *groov* Box inside the machine, accessed locally using SoftAP.



Track machine data

It's often important to track data about the machine while it is installed at a customer's site, for purposes of billing, tracking machine performance, predictive maintenance, and so on. With the *groov* Box in the machine and connected to a network with internet access (using its second network interface), you can use the included Node-RED to communicate machine data to software either locally or in the cloud for tracking, logging, and analysis.

To avoid having to ask customers to set up a VPN or change their firewall for this purpose, use the Ignition Edge MQTT transport protocol with Sparkplug messaging to publish machine data to a broker. With either Node-RED or MQTT, all communications from the *groov* Box are outbound, so no changes are required to the customer's firewall.



SETTING UP A VIRTUAL PRIVATE NETWORK (VPN)

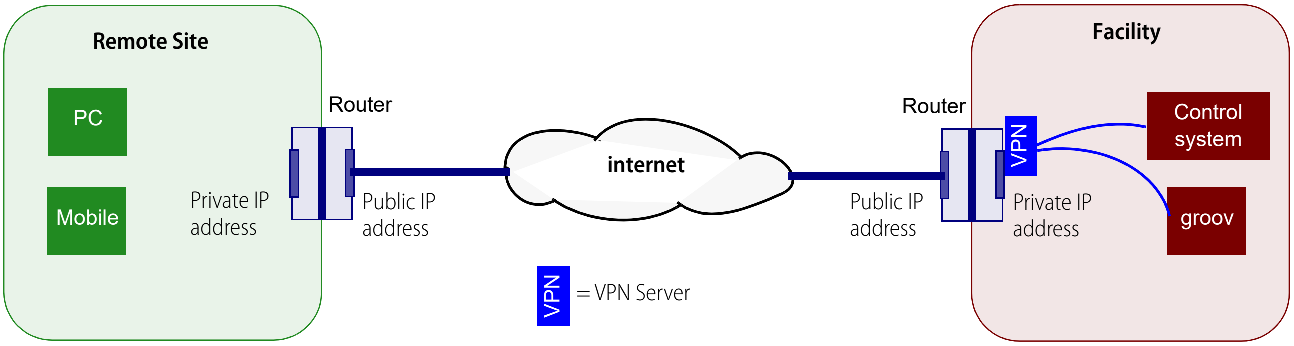
If your application requires a VPN, in most cases you will need to set up a VPN server on your network and set up VPN clients. If you have an IT department, work with them to do both.

Setting up a VPN server

If you don't have an IT department, you can google for ways to set up your own VPN server. You can install VPN software on a network device or a computer, for example, a Microsoft Windows Server machine configured for VPN Server. Several protocols are available for VPN, including IPsec, OpenVPN, and the older PPTP (point-to-point tunneling protocol), which is generally not considered as secure as OpenVPN or IPsec.

Choose the VPN protocol based on what your VPN clients (your PCs and mobile devices) need to use. Note that PPTP is no longer recommended by Apple, and support stopped as of iOS 10/macOS Sierra. Apple recommends using IPsec or SSL VPN clients on the App Store.

Place the VPN server inside your private network, behind the router, at your facility:



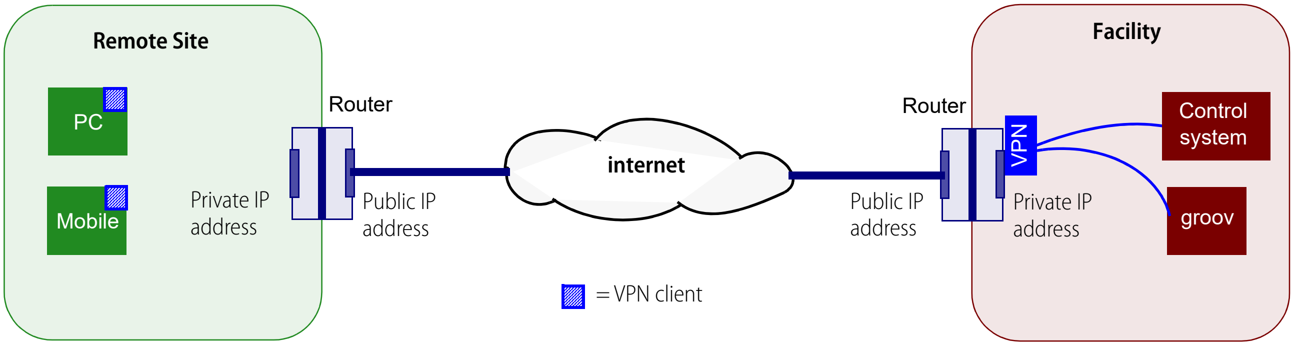
In the router, set up a port forward rule so the router will know to send data through the proper port to reach the VPN server. Port numbers depend on the VPN protocol you're using:

VPN protocol	Ports used
PPTP	47 and 1723
IPsec	50, 51, and 500
OpenVPN	1194

On the VPN server, set up users and accounts so authorized individuals have usernames and passwords to use the VPN.

Setting up VPN clients

Once your VPN server is set up and user accounts established for those who need them, you'll need to set up a VPN client on each PC or mobile device that will use the VPN. In the diagram below, a PC and a mobile device are shown at the same remote site, but they can be anywhere:



NOTE: In our diagram we're assuming that the control system and groov are at the same location. If they're not, and you need to have groov communicate with a remote PAC, see "VPN special case" on page 24.

Most current PCs and mobile devices have VPN client software built in. The VPN client must use the same VPN protocol (for example, IPsec or OpenVPN) as the VPN server. Some clients give you a choice.

To set up VPN clients, follow the steps below for the devices you're communicating with:

- VPN: Computer to groov—page 23
- VPN: Android mobile to groov—page 23
- VPN: iOS mobile to groov—page 24

VPN: Computer to *groov*

1. Make sure you have a VPN account on the VPN Server.
2. Set up a VPN client on your computer.
   - For Windows 10, see: https://support.microsoft.com/en-us/help/20510/windows-10-connect-to-vpn
   - For macOS Sierra, see: https://support.apple.com/kb/PH25513?locale=en_US&viewlocale=en_US
3. Establish a VPN connection to your VPN server from your PC.

   When you connect, the remote VPN network assigns your PC an additional IP address to match the local network.
4. To use *groov*, in your web browser, type https:// plus the hostname or IP address of the *groov* Box (or the PC running *groov* Server for Windows). Example: `https://mygroov`
5. Test your connection: see "Testing communication" on page 26.

VPN: Android mobile to *groov*

On an Android device, follow the steps here to set up a VPN client:
http://www.howtogeek.com/135036/how-to-connect-to-a-vpn-on-android/

Use the steps under *Integrated VPN Support*. It is not necessary to install any third party apps or root your phone like other sections of the article mention.

For a VPN server running PPTP, choose the following:

- VPN server running PPTP
- VPN Server address
- DNS search domains

Leave everything else at the default selection.

When connecting to the VPN, enter your username and password.

Once connected, open a web browser and type https:// plus the hostname or IP address of the *groov* Box (or the PC running *groov* Server for Windows). Example: `https://172.20.52.5`

If you have problems connecting, see "Testing groov from inside your facility" on page 26.

For an OpenVPN server, visit the Google Play Store and download the OpenVPN Connect app: https://play.google.com/store/apps/details?id=net.openvpn.openvpn&hl=en

Configuration is slightly different, because OpenVPN uses IPsec and requires a certificate (generated by the administrator of the OpenVPN server) to be installed on your phone. Work with your IT department or OpenVPN administrator to install the certificate.

VPN: iOS mobile to *groov*

NOTE: You cannot use a VPN server running PPTP on iOS 10 or higher.

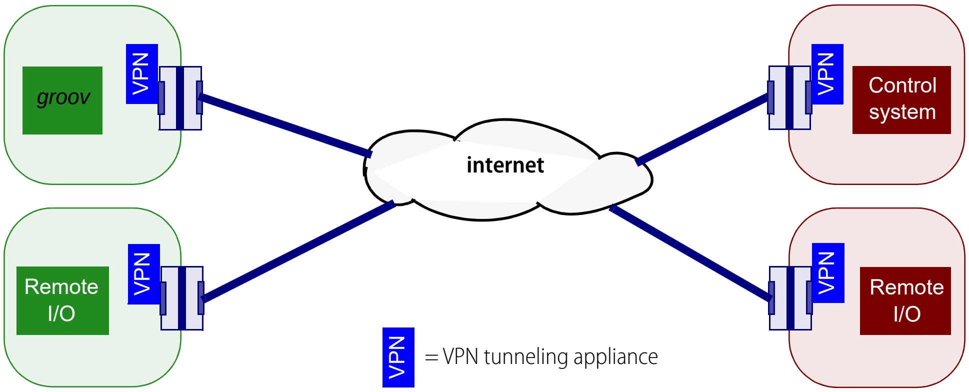
1. Go to the iTunes store and download an SSL VPN client app (such as OpenVPN Connect).
2. When connecting to the VPN, enter your username and password.
3. Once connected, open a web browser and type https:// plus the IP address of the *groov* Box or the PC running *groov* Server for Windows. Example: `https://172.20.52.5`

   If you have problems connecting, see "Testing groov from inside your facility" on page 26.

VPN special case

If your *groov* Box is in a different location from your control system, you'll probably need to set up the VPN in a different way: by using a **VPN tunneling appliance** rather than separate VPN server and VPN client. The *groov* Box, as well as many control systems and equipment, cannot be VPN clients nor accept a VPN client. And it is unsafe to use port forwarding unless the controller or device has built-in user authentication.

A VPN tunneling appliance solves the problem by incorporating both server and client in one box, which may or may not also include a gateway router. You place a VPN tunneling appliance at each end of the communication to create a tunnel that can go both ways. The VPN tunneling appliances create a site-to-site VPN, not just a device-to-device VPN. Once you set up the two appliances, *groov* and your control system communicate just as if they were on the same local area network.



USING PORT FORWARDING (PF)

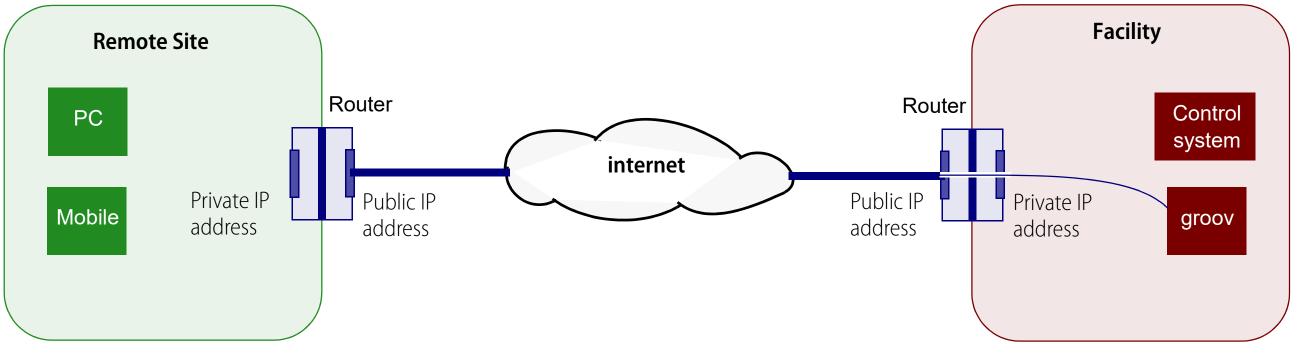
Port forwarding is a less secure way than a VPN to communicate between a PC or mobile device and *groov*, but it may be an acceptable option if:

- You have no IT department.
- You have no VPN server and do not want to set one up yourself.
- You have just one *groov*.

CAUTION: *Port forwarding is NOT recommended for any communication with a SNAP PAC controller.*

In "internet gateway routers" on page 16 we saw how communication happens over the internet. Your private network is hidden from the public internet behind a gateway router (a firewall) that shows only its public IP address to the world.

A port forward rule creates a small hole in that firewall to allow data packets to get through to the private network. PF is less secure than a VPN because a VPN creates a layer of authentication and encryption which may or may not be present with PF. PF can work with *groov* because *groov* provides encryption and requires user authentication with usernames and passwords; unauthorized users cannot get in.



Port forwarding creates a hole in your firewall to allow certain packets through. PF works for *groov* because *groov* provides encryption and requires authentication from all users.

For more information on port forwarding, see: http://portforward.com/

To use port forwarding, you need to:

- Determine the IP address (see "Determining the IP address," below).
- Establish a port forward rule on the gateway router (see page 25).
- Set up port forwarding on the client PCs or mobile devices (page 26).

Determining the IP address

On the private network where *groov* is, find out the gateway router's public IP address (see steps in "Gateway router identification" on page 17). In the diagram above, that's the facility's router, which gives access to *groov*. You'll use this address to reach your devices, or you'll map a domain name to it.

Determine whether you will use the router's IP address or its domain name for your connection (see "Fixed (static) vs. dynamic IP addresses" on page 18).

Establishing port forward rule(s)

By default, *groov* communicates over port 443 or 8443.

NOTE: Sometimes changing the port number is recommended, but because changed port numbers are easy to discover, this "security by obscurity" suggestion provides no real extra security. If you make a port forward rule that port 4321 goes to port 443 of the groov Box's IP address, you'll need to always add the port number when you access your groov Box.

If you have more than one *groov* Box behind the router/firewall, you must configure a port forward rule in the router for each one. Each rule has to have a different port number going to a different IP address. Otherwise the router can't differentiate between them.

Avoid using the reserved port numbers listed in "Opto 22 port usage" on page 20. For more general information on ports, see "port" on page 31.

Creating the port forward rule

You'll need to know the following:

- How to access the web page or configuration software for your router/firewall, including its username and password
- IP address of the *groov* Box (If your router requires an IP address for port forwarding, assign a fixed IP address to your groov Box.)

1. Open the webpage or configuration software program for the router/firewall.
2. Locate the link to create the port forward rule.
3. Create a rule that says any internet traffic coming in on a specific port number should be sent to the IP address and port number of the *groov* Box.
   – Use the default port number 443 (or 8443), unless you have changed it, have more than one *groov*, or want to obscure the port number by setting it to something else.
   – If given an option to apply the rule to TCP or UDP or both, make the rule apply to both.

Setting up port forwarding on PCs and mobile devices

PF: PC or mobile to *groov* using a web browser

If you've already set up the port forward rule on your router or firewall, you're ready to test communication: see "Testing groov from inside your facility" on page 26.

PF: Mobile to *groov* using the *groov* View mobile app

1. On an **iOS** mobile device, go to Settings > groov.
   On an **Android** mobile device, launch the *groov* View app and tap Connect to groov.
2. In the URL field, enter https:// and the hostname or IP address of the router that has the port forward rule.
   Example: `https://myrouter`
3. In the Port field, enter the port number you used when setting up the rule (usually 443).

   You're ready to use the app. See the *groov Build and View User's Guide* for instructions.

TESTING COMMUNICATION

You're now ready to test communication between your two networks.

Testing *groov* from inside your facility

Since you've set up your VPN or port forward rule while on your local network, you're likely to want to test it from there as well. But testing an outside connection from inside may not work. internet service providers (ISPs) often won't allow communication to go outside and then come right back in.

To test the connections you've set up from inside, don't use your computer. Instead, use your smartphone or another mobile device with cellular service.

1. Turn off WiFi to force the phone to connect with the nearest cell tower and thus be outside your local network.
2. **VPN:** Open a web browser and type https:// plus the *groov* Box's hostname (or its fixed IP, if you assigned one).

   VPN example: `https://mygroov`
   or `https://10.162.89.1`

PF: Open a web browser and type https:// plus the IP address or domain name of the router for the *groov* network, plus a colon and the port number.

PF example: `https://203.208.65.21:443`
or `https://mydomain:443`

Testing *groov* from outside your facility

To connect with *groov* from outside your LAN, use either a computer or mobile device.

- **VPN:** Open a web browser. For the URL, type https:// plus the *groov* Box's hostname (or its fixed IP, if you assigned one).

  VPN example: `https://mygroov`
  or `https://10.162.89.1`
- **PF**: Open a web browser. For the URL, type https:// plus the IP address or domain name of the router for the *groov* network, plus a colon and the port number.

  PF example: `https://203.208.65.21:443`
  or `https://mydomain:443`

Troubleshooting

If you have any problems connecting, see the *groov Build and View User's Guide*.

If you've been able to communicate with your *groov* in the past and suddenly receive timeouts and can't connect, your IP address has likely changed. That's why you need to use a domain name and a DDNS. See "Fixed (static) vs. dynamic IP addresses" on page 18.

IMPORTANT: *On your groov Box, make sure that you configure only ONE of its interfaces with a valid gateway and DNS IP address, and connect that interface to the gateway router that's connected to the internet. Set the other interface's gateway and DNS addresses to 0.0.0.0. If more than one interface has a gateway configured, the groov Box won't know which one to use for communications.*

4: Glossary and Resources

NETWORKING TERMS

This short glossary includes some of the networking terms and concepts we use in this guide. For a lot more information, search the internet for these terms and any others you're not sure about.

client

In computer networking, a *client* requests data or services that are then supplied by a server on the same network. A client is typically a software program. For example, a client such as Microsoft Word might request a print server on the network to print a Word document.

DHCP

DHCP (dynamic host configuration protocol) helps devices on a network communicate with each other. A DHCP server uses the protocol to assign an IP address and other configuration information to each device as soon as it appears on the network.

Because these assigned IP addresses are valid only for a certain length of time, the address of a specific device on the network is likely to change over time and is referred to as dynamic. (In contrast, a fixed or static IP address is permanently assigned to a device and will not change.)

DNS/DDNS

DNS (domain name system) is a service that resolves domain names (like `google.com`) or computer names (like `//mypc`) into IP addresses. Typically the DNS service is provided by a computer or pub-sub.

Communication between computers and other devices on a network is based on IP addresses; each address is a series of numbers. A DNS is useful because humans cannot remember numbers as easily as they can remember words.

A *DDNS (dynamic domain name service)* updates domain names in the DNS that have dynamic (changing) IP addresses. Most IP addresses change over time; a DDNS periodically checks and sends the change to DNS servers.

domain

A *domain* is a group of computers accessible via fully qualified hostnames that contain the same domain name. The *domain name* usually reflects the company's or organization's name so it is easy for people to remember when they want to access it over the internet.

A company like Opto 22, for example, has a domain that's used for all internet communications. Opto 22's domain name is opto22.com.

gateway

Gateway is a general term that refers to a means of providing access to a place or to data. A router may be called a gateway, especially when it provides access to the internet.

IP address

An *IP address* is a numeric address assigned to a computer or other device on a network that uses the Internet Protocol (IP) for communication. An IP address identifies a device and provides a location for communication. The more familiar IPv4 addresses are in the format of four decimal numbers (values 0–255), separated by dots. For example: `192.168.10.4` or `10.172.0.244`

IPv6 addresses (which are becoming increasingly more common) are formatted in eight groups of four hexadecimal digits, separated by colons. For example:
`2001:0db8:0000:0042:0000:8a2e:0370:7334`

LAN

A *LAN* is a local area network, usually a private network set up by an individual, a business, or an organization to connect computers and other electronic devices within a limited physical area. Compare to WAN.

MQTT

MQTT (at one time called Message Queue Telemetry Transport, although it does not do queuing) is a lightweight communication protocol based on the pub-sub architecture. It is often used for internet of things (IoT) applications because it is suitable for data communication with remote devices that don't have much computing power or are on networks with irregular connections or low bandwidth.

MQTT with Sparkplug messaging (which validates data sent over MQTT and makes sure it is current) is available with Ignition Edge in the *groov* Edge Appliance.

network

A *network* is a group of computers or other electronic devices linked together so they can exchange information. The link requires some form of physical connection, usually through wires or airwaves, and a common *protocol*, which is a language through which information is exchanged.

This guide covers Ethernet networks and wireless networks. It does not include information about serial or other kinds of networking with Opto 22 products.

network switch

A *network switch* directs data traffic between the devices connected to it. The switch transmits data from one device to another using the device addresses. In contrast to a *hub*, which transmits any communication to all devices on the network, a switch transmits only to the specific device the data is addressed to.

node

An individual computer or other device on a network is called a *node*.

port

One device can communicate in a number of different ways using the same IP address and transport protocol. For example, a *groov* Box can communicate with Modbus/TCP devices, a SNAP PAC controller, and an OPC UA server, all at once using the same IP address and protocol.

Each of these "services" uses a unique protocol and *port* number combination (for example, TCP 443 or UDP 443) for communication. The combination of IP address/protocol/port number keeps communication running smoothly. It's like an apartment building where all the apartments have the same street address (IP address and protocol), but each apartment has a number (the port number).

Generally ports 0 to 1023 are well-known ports and should not be used for anything other than their assigned service. For example, port 80 is used for HTTP (web communication), port 25 is used for email, and port 21 is used for FTP (file transfer protocol).

Ports 1024 to 49151 are registered ports. Many of these have been assigned to specific companies to use for their specific services. For example, ports 22000–22005 are registered to Opto 22. But many port numbers between 1024 and 49151 are available for use by anyone.

Official port assignments are maintained by IANA, the Internet Assigned Numbers Authority.

port forwarding

Port forwarding allows remote computers (for example, computers on the internet) to connect to a specific computer or service within a private local-area network (LAN).

Port forwarding opens certain ports on your home or small business network, usually blocked from access by your firewall, to the internet.

pub-sub

A *pub-sub* (or *publish-subscribe*) architecture differs from a request-response architecture in ways that make it useful for internet of things (IoT) applications.

In pub-sub, all data is held by a broker, which may be located on your network or in the cloud. Devices and software publish data to the broker, or subscribe to data the broker holds, or both. Data is published only when it changes (report by exception) and sent to subscribers only when it changes.

request-response

Request-response (also called command-response or query-response) is a basic communcation method among computers on a network. One computer sends a request for data or services, and another responds by sending the requested data or performing the service. The computer that sends the request is the client, and the computer that responds is the server.

router

A *router* is a networking device that lets packets of information from one network end up on another. The router is connected to two or more networks. When a data packet arrives at the router, the router checks its IP address and forwards it based on established rules kept in a routing table.

Routers may allow communication between private networks, for example two LANs in the same business, or between a private network and the internet (a LAN and a WAN).

server

In computer networking, a *server* shares resources and data among clients on the network. The server provides data or services when requested by a client.

For example, print servers manage and allocate printer resources for a network; file servers store and allow access to folders and files needed by multiple users on a network; web servers present web pages to clients like PCs, tablets, and smartphones.

softAP

A software-enabled access point, or *softAP*, is software that turns a computer device into a WiFi access point. The typical use is to allow other devices to access the internet through the device that has softAP. A softAP can also be used to create a separate WiFi network not connected to the internet, as in the *groov* Box.

Sparkplug

An open messaging system developed by Cirrus Link Solutions that defines topic namespace, session state management, and data payload encoding for devices and software using the MQTT transport protocol. Sparkplug adds specific features designed to standardize MQTT messages for industrial applications.

For more information, see the Sparkplug specification.

subnet mask

The *subnet mask* defines the IP address range of a local area network, or LAN. A subnet mask is a way of logically segmenting a network, limiting access to specific IP addresses unless communication passes through a router. All devices with the same network prefix (calculated by a bitwise AND between the subnet mask and the IP address) are on the same LAN or subnet.

When you configure a device on the network, you assign a subnet mask together with the IP address. If you assign a fixed IP address to a *groov* Box, you also enter the subnet mask.

The subnet mask and the IP address work together, a little like a country code on the phone. You add the country code to the phone number, and the system uses that information to connect you. The most common subnet mask is `255.255.255.0`. In this mask the first three parts identify the network, and the last part identifies the node or host. For this subnet mask, all devices on the network would have addresses between `192.168.1.0` and `192.168.1.254`*.

Network	**Node**
`255.255.255.`	`0`
`192.168.1.`	`0`
`192.168.1.`	`254`

* The last address (x.x.x.255) is reserved for subnet-directed broadcasts.

VPN (virtual private network)

A *VPN (virtual private network)* is a method of connecting computers or other devices remotely, over the internet, as if they were on a private local area network (LAN). A VPN provides a kind of shielded tunnel through the internet, maintaining private security and encryption.

From the user's point of view, the VPN makes it feel as though he were right there on the same private network. VPNs are often used for employees who are traveling or working at remote sites.

WAN

A *WAN* is a wide area network, which may be private or public. The internet is the prime example of a public WAN. Compare to LAN.

RESOURCES

These are just a few of the many resources online that deal with remote networking.

Information from Department of Homeland Security, Industrial Control Systems Cyber Emergency Response Team: Recommended Practices for Industry

Networking FAQs plus a forum for asking questions: Whatismyip.com

Some DDNS services:

- http://dyn.com/dns/
- http://www.noip.com/

MQTT:

- MQTT 101: How to Get Started
- mqtt.org
- Sparkplug messaging
- Sparkplug specification
- List of MQTT brokers for testing or prototyping

Additional information about VPNs:

- An introduction to VPNs and how they work
  http://www.rawbytes.com/virtual-private-networks-in-depth-technical-details/
- Microsoft technical information for Windows Server 2016 and Windows 10
  https://docs.microsoft.com/en-us/windows-server/remote/remote-access/vpn/vpn-top

Setting up a VPN:

- On Android: http://www.howtogeek.com/135036/how-to-connect-to-a-vpn-on-android/
- Download the OpenVPN Connect app for Android:
  https://play.google.com/store/apps/details?id=net.openvpn.openvpn&hl=en
- For macOS Sierra: https://support.apple.com/kb/PH25513?locale=en_US&viewlocale=en_US
- For an iOS mobile device, download the OpenVPN Connect app for iOS from the App Store.

Information about port forwarding: http://portforward.com/