

# Venus OS Large image: Signal K and Node-RED

## 1. Introduction

This manual documents Venus OS (Operating System) Large. Venus OS (Operating System) is the software running on our Cerbo GX monitoring device <https://www.victronenergy.com/panel-systems-remote-monitoring/cerbo-gx> as well as other GX devices. Venus OS (Operating System) Large is an extended version of the common software, adding Node-RED <https://nodered.org/> and Signal K server <https://www.npmjs.com/package/signal-k-server>.

Node-RED offers PLC like programming for connecting hardware devices, APIs and online services. It provides a browser-based editor that makes it easy to wire together flows. With it, one can for example program a functionality such as activating a relay based on a temperature measurement. Or make far more complex algorithms, tying relays, measurements, or other data available from Venus OS (Operating System) or elsewhere together. All without having to write real source code, as Node-RED calls Low-code programming for event-driven applications.

Also Node-RED features a fully customisable dashboard <https://flows.nodered.org/node/node-red-dashboard>, viewable in a webbrowser - both locally but also remotely, via the VRM Servers.

Signal K server is aimed for yachts, and multiplexes data from NMEA0183, NMEA 2000, Signal K and other sensor inputs. It makes all that data available for Apps, such as WilhelmSK <https://www.wilhelmsk.com/>, a highly customizable boat instrument display. The Sailing with Signal K blogpost on our website <https://www.victronenergy.com/blog/2019/06/13/sailing-with-signal-k/> is a great read for more information and ideas.

## 2. Where to find support

There is and will be no official support on the functionalities of Venus OS (Operating System) large.

For support, turn to the Modifications Space on Victron Community <https://community.victronenergy.com/spaces/31/index.html>.

There, first search if someone else had a similar question already. And if you can't find such, then create a new question.

For support on SignalK, there is a #victron channel in the SignalK Slack <http://slack-invite.signal-k.org/> workspace.

## 3. Requirements

### 3.1 GX Device compatibility

The compatible GX devices are:

- Cerbo GX
- Cerbo-S GX
- Venus GX
- Octo GX
- MultiPlus-II GX
- EasySolar-II GX

The Color Control GX (CCGX) as well as the CanVU GX are unfortunately not able to run Venus OS (Operating System) Large.

Venus OS (Operating System) large is also available for certain RaspberryPi models. Installation files below, and installation & other raspberrypi specific instructions [here https://github.com/victronenergy/venus/wiki/raspberrypi-install-venus-image](https://github.com/victronenergy/venus/wiki/raspberrypi-install-venus-image).

### 3.2 CPU & Memory resources

Both SignalK as well as Node-red are rather resource intensive software applications to run on the GX Device. From a CPU usage point of view as well as RAM memory.

Ofcourse, how intensive depends on what kind of flows (node-red) or how many messages & handling of them (SignalK) happens in the system.

From a GX device point of view, from the support models the Venus GX is the one with the least resources. Running both SignalK and Node-RED at the same time will be "pushing it" to say it simply. The Cerbo GX is already much faster, as it has a dual core CPU as well as much more RAM. More details about CPU & RAM differences are in this document: <https://github.com/victronenergy/venus/wiki/machines> <https://github.com/victronenergy/venus/wiki/machines>.

When running out of RAM or CPU, the device will slow down. And eventually also reboot by itself due to the watchdog.

## 3. How to install and update

### Step 1. Preparations

- [Cerbo GX](#) - No preparations required.
- [Venus GX](#) - First install & boot v2.73, or a later version. And only thereafter install Venus OS (Operating System) Large. The reason for this is to make sure the partitioning of the flash is per requirements of Venus OS (Operating System) Large.
- [MultiPlus-II GX](#) - No preparations required.
- [EasySolar-II GX](#) - No preparations required.
- [RaspberryPi 3](#) - No preparations required.
- [RaspberryPi 4](#) - No preparations required.

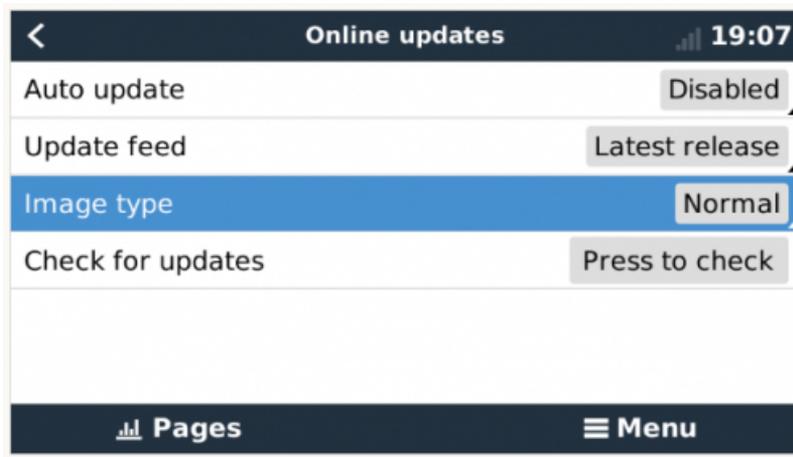
Note, when installing on a RaspberryPi, do make sure to familiarise with the Venus OS for raspberrypi documentation <https://github.com/victronenergy/venus/wiki/raspberrypi-install-venus-image> as well.

### Step 2. Update to the latest firmware version (v2.90 or later)

See Settings → Firmware → Online updates. For more information, refer to the user manual of your GX device.

### Step 3. Set the image type Venus OS Large

Go to Settings → Firmware → Online updates → Image type:

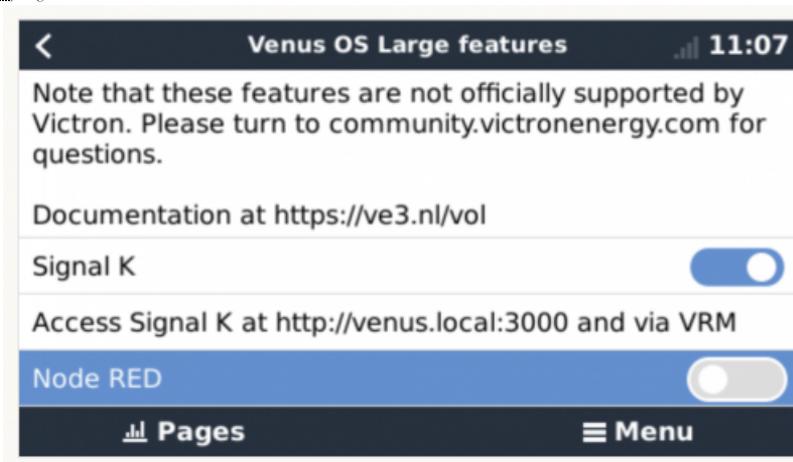


Now, change the Image type to Large. Then go one menu up, and Check for a new version and install that.

### Step 3. Verify and enable

1. Verify that the image was successfully installed: go to the Settings menu, and check that the Venus OS (Operating System) Large menu is available.
2. Enable either Signal K or Node-RED. Note that though its possible to enable both, doing that on a Venus GX is "pushing it", the system might become slow. This is less of a problem on a Cerbo GX, as it has more resources. Installer access level is required to enable/disable these settings.

Visible once successfully installed, the Venus OS (Operating System) Large menu:



### How to update

Both online and offline update methods can be used. The Image type setting, see the How to install chapter, defines what image type is being updated. See the GX manual for more details.

## 4. Accessing Node-RED and Signal K

After installing and then enabling either one or both of the features, they need to be configured through a website in your browser.

To access the Node-RED flow editor, type <https://venus.local:1881/> [<https://venus.local:1881/>]. Note that it might be necessary to replace venus.local by an IP address, for example like this: <https://192.168.178.224:1881/> [<https://192.168.178.224:1881/>]. It is normal for the browser to show a security warning. Proceed according to browsers instructions.

See the Remote Console on LAN chapter on our GX manual [[https://www.victronenergy.com/media/pg/Cerbo\\_GX/en/accessing-the-gx-device.html#UUID-a007c0de-85f1-a53f-550b-250ed46ac1f](https://www.victronenergy.com/media/pg/Cerbo_GX/en/accessing-the-gx-device.html#UUID-a007c0de-85f1-a53f-550b-250ed46ac1f)] for how to find the IP address.

To access the Signal K admin panel, type <http://venus.local:3000/> [<http://venus.local:3000/>]. Just like Node-RED above, note that it might be necessary to replace venus.local in that URL (Uniform Resource Locator) by an IP address. For example like this: <http://192.168.178.224:3000/> [<http://192.168.178.224:3000/>]

These two screenshots show what you will see after successfully accessing the pages:

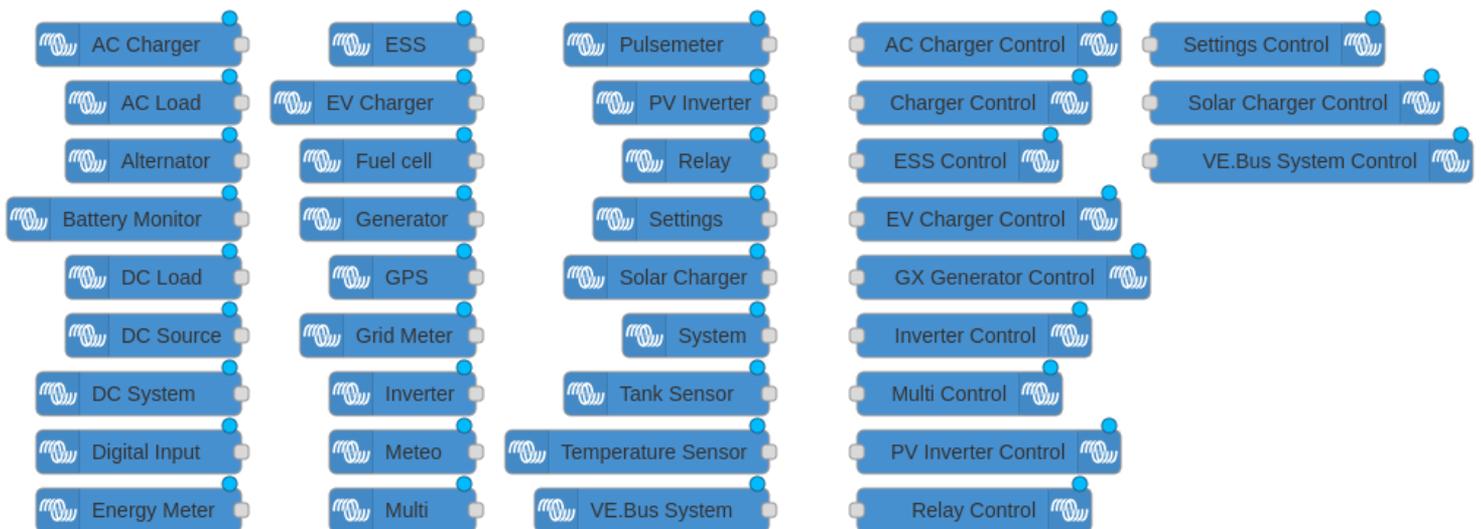
### Node-RED flow editor

The image shows a web browser window displaying the Node-RED interface. The browser's address bar shows the URL `https://192.168.178.224:1881/#flow/3fc4fabb266c4e70`. The Node-RED header includes a "Deploy" button and a user profile icon. The main workspace is titled "Flow 1" and is currently empty. On the left, the "filter nodes" sidebar is open, showing two categories: "common" and "function". The "common" category includes nodes like inject, debug, complete, catch, status, link in, link call, link out, and comment. The "function" category includes function, switch, and change nodes. On the right, the "info" panel is active, displaying details for a node titled "Hydraulic battery | Voltage (V)". The node's ID is `"37064004cc398820"` and its type is `victron-input-battery`. Below the node details, there is a note: "Switch flow tabs with `ctrl-[` and `ctrl-]`".

Signal K admin panel

Signal K Server version 1.44.0 Sponsor Signal K - Venus

## 5. Node-RED



### 5.1 Introduction

Node-RED is a tool for connecting hardware devices, APIs and online services in new and interesting ways. It provides a browser-based editor that makes it easy to wire together flows. With it, one can for example program a functionality such as activating a relay based on a temperature measurement. Or make far more complex algorithms, tying relays, measurements, or other data available from Venus OS (Operating System) or elsewhere together. All without having to write real source code, as Node-RED calls Low-code programming for event-driven applications.

Also Node-RED features a fully customisable dashboard <https://flows.nodered.org/node/node-red-dashboard>, viewable in a webbrowser - both locally but also remotely, via the VRM Servers.

A good tutorial on how to work with Node-RED is here: <https://nodered.org/docs/tutorials/first-flow>. On <https://github.com/victronenergy/node-red-contrib-victron/wiki/Example-Flows> you can find example flows that use the Victron Energy nodes. Searching for Victron <https://flows.nodered.org/search?term=victron> will give all community flows that have been tagged "victron".

There is also a 3 part introductory webinar on Node-RED and Venus available at the Victron Tech YouTube channel:

1. Getting started [https://www.youtube.com/watch?v=i\_jaciqn\_FgI]
2. Basic building blocks and the dashboard [https://www.youtube.com/watch?v=J2vNLEdIq]
3. Complex Flows and dashboard example [https://www.youtube.com/watch?v=8e2MBo5seM]

The Node-RED editor can be accessed from your LAN (Local Area Network) at <https://venus.local:1881/> [https://venus.local:1881/]. For some systems, you'll have to replace `venus.local` by the ip-address. You can also access the Node-RED editor from VRM, where it is available via the *Venus OS (Operating System) Large* menu.

**Standard nodes + Victron nodes** Node-RED comes with a core set of useful nodes, and within the Venus OS (Operating System) large install there is a set of Victron nodes added to that: the Victron Palette. All of the different nodes are described here [https://github.com/victronenergy/node-red-contrib-victron/wiki/Available-nodes].

For developers: source code for the node-red-contrib-victron modules is here [https://github.com/victronenergy/node-red-contrib-victron/], and note that updating is not possible from either the Node-RED editor or as the commandline. The recommended way to get to the new node-red-contrib-victron version is to await a new Venus OS (Operating System) large build.

## 5.2 Back-up and restore

Just as with other software it is important to keep your Node-RED flows save and secure. We advise you to periodically backup your flows using the palette manager “Export → all flows” option to download the `flows.json` file to your PC.

On a fresh installation you can then use the “Import” option to import this file. Note that importing flows that rely on unavailable nodes will complain about that. See below in how to install extra nodes.

It is always wise to create an extra backup before performing an update of the firmware.

Do note that the above instruction only makes a backup of your flows, not of the Node-RED configuration. The configuration is stored in `.config/*.json` files under `/data/home/nodered/.node-red/`.

## 5.3 Installing extra nodes

Node-RED comes with a core set of useful nodes, but there are many more available from both the Node-RED project as well as the wider community.

For example nodes that give access to weather data, or with timer functionality, and so forth. You can find several example flows on <https://github.com/victronenergy/node-red-contrib-victron/wiki/Example-Flows> [https://github.com/victronenergy/node-red-contrib-victron/wiki/Example-Flows]

You can search for available nodes in the Node-RED library.

The Victron Palette comes pre-installed.

Read the [Adding nodes to the Palette - guide by Node-RED](https://nodered.org/docs/user-guide/runtime/adding-nodes) [https://nodered.org/docs/user-guide/runtime/adding-nodes] to learn how to install and manage node palettes.

To install new nodes/palettes requires a working internet access.

Palettes will be installed on the data partition on the GX Device. Which means that they will remain installed when updating Venus OS (Operating System) to a new version.

Refer to the remove button in the palette manager to remove self-installed palettes.

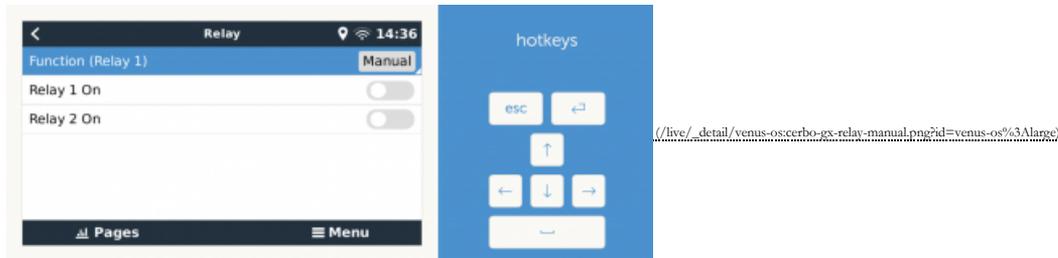
When installing a newer version of a palette than what is pre-installed in Venus OS (Operating System), Node-RED will use that newer version.

Make sure that there is always 30MB of free space remaining on the data partition. One of the [example flows](https://github.com/victronenergy/node-red-contrib-victron/wiki/Example-Flows#monitor-disk-usage) [https://github.com/victronenergy/node-red-contrib-victron/wiki/Example-Flows#monitor-disk-usage] shows how to check the free space using a Node-RED dashboard.

## 5.4 Controlling relays

### GX Device relays

To be able to control the relay(s) on the GX Device, configure them to be Manually controlled:



### Battery Monitor Relay

The relay in a BMV can be controlled. Make sure to set, using VictronConnect, its option in the BMV configuration to REM, which is short for Remote Controlled.

The relay in a Lynx Shunt VE.Can can be controlled. Make sure to set the relay function to “Manual control” in the user interface of the GX.

The Relay in the Lynx Smart BMS is not available for control via Node-RED.

### SolarCharger MPPT Relay

Currently it is not possible to control the relay in a MPPT Solar Charger, even though there is a Victron node offering such functionality. The reason it doesn't work, is that VictronConnect does not have the required “Manual control” option in the list of selectable Relay modes for the MPPTs. This will be fixed at some point.

Note that only the larger model solar chargers feature a relay.

### (VE.Bus connected-) Multi, Quattro and similar product relays

Its not possible to control the alarm relay nor the AC-out 2 contactor of our inverter/chargers. Also, there are no plans to make them controllable remotely, which includes no plans to make them controllable from within Node-RED either.

### RS series (inverter, multi) relays

Currently not controllable. Will be controllable in the future.

### Charger relays

The only charger which has a relay that is controllable is the Phoenix Smart Charger. To use that for remote control, set its mode, using the VictronConnect app, to be remote controllable. Note that its use is limited: the relay is only powered (and thus only controllable) when the charger is switched on. It doesn't work when it's turned off with AC connected. This limitation applies to the 230V models as well as the later 120V/240V models.

## 5.5 Node-RED Dashboard

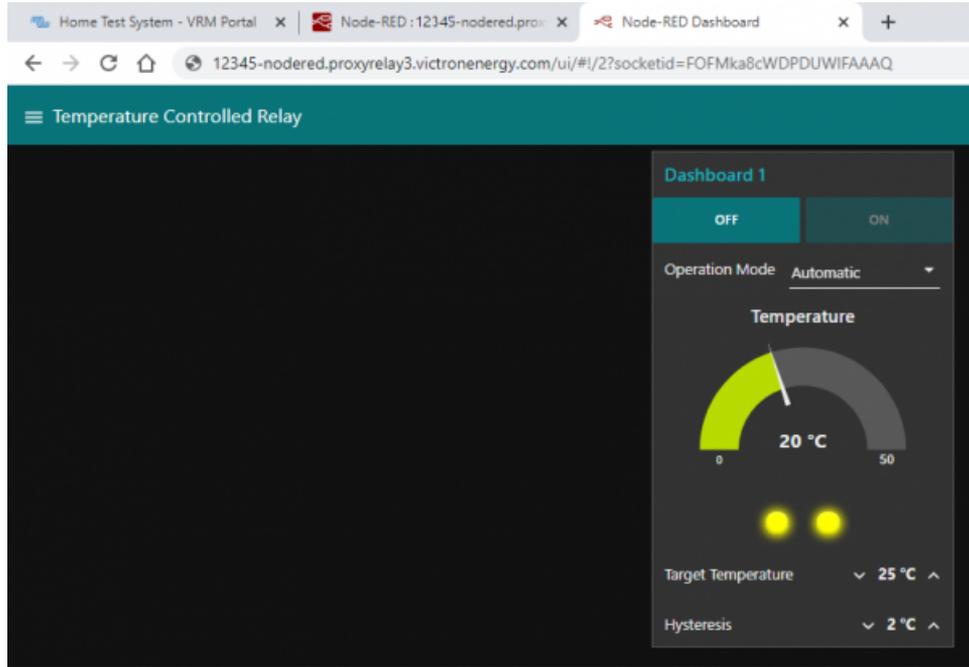
Besides the flow and automation functionality, Node-RED also offers a fully customisable dashboard. Including graphs, indicators as well as controls such as switches and sliders.

It is viewable in a web browser - both locally as well as remote, via the VRM Servers. See chapter 6.2 for how to see it remotely.

The dashboard needs to be installed, just like other extra palettes, using the Manage Palette feature, on the right of the screen in the menus.

Once that is done, you can find the nodes in the palettes on the left.

Example dashboard:



[/live/\\_media/venus-os:vrn\\_node\\_red\\_ui\\_step\\_3.png](#)

## 5.6 Configuration files

WARNING: the vast majority of systems using Node-RED will not, and should not (!), have to modify any of the files here described. Venus OS (Operating System) itself, including Venus OS (Operating System) large and Node-RED, is made such that its not necessary to dive into the command line.

The information here provided is for the benefit of the (few) users that do need or want to change them.

On startup of Node-RED on Venus reads three settings files.

1. `./usr/lib/node_modules/node-red/venus-settings.js` is read first. Never change this file as it will be overwritten by a Venus OS (Operating System) update.
2. `./data/home/nodered/.node-red/settings-venus.js` is used to override and/or extend the first file and will survive Venus OS (Operating System) updates. Do not edit this file as this is for Victron to use.
3. `./data/home/nodered/.node-red/settings-user.js` is used to override and/or extend the second file and will survive Venus OS (Operating System) updates. You are allowed to edit this file if you please, but don't be too surprised if that causes Node-RED to break after an upgrade.

When using these files, make sure that the configuration is enclosed within a `module.exports = {}` block.

There are more configuration files in `./data/home/nodered/.node-red/`. You might not have all of them on your system or even some more if you installed one or more of our candidate releases.

- `flows.json` - This is the flowFile, which is used to store your flows. You can also export this file from the Node-RED menu.
- `flows_cred.json` - If your flows contain credentials/login information, these get stored in this file.
- `.config.*` - These files are for Node-RED to store its settings.

For more information on such Node-RED configuration customisations, see <https://nodered.org/docs/user-guide/runtime/configuration> <https://nodered.org/docs/user-guide/runtime/configuration>.

## 5.7. Node-RED FAQ

### Q: Write/read file gives "Error: EACCESS: permission denied"

Since Venus OS (Operating System) 2.90 and onwards, Node-RED no longer runs as user `root`, but as user `nodered`. This means that the Node-RED flows are only allowed to access files and directories that are owned by the `nodered` user. These files typically start below `./data/home/nodered/`. So modifying the storage location to something like `./data/home/nodered/storage.txt` ought to work.

### Q: How can I update the Victron node palette

As that palette is part of the total Venus OS (Operating System) large package, it can only be updated by installing a newer Venus OS (Operating System) large version.

In case you see that there is a newer version Victron palette available online, and thats not yet shipping in a Venus OS (Operating System) Large version, then the only option is to wait: a new Venus OS (Operating System) Large version is usually made available shortly after updating that palette.

### Q: How can I password protect Node-RED

Password protection of Node-Red is linked to the remote console. If a password is set for the remote console, this password is also used for securing Node-Red. The username is `admin`, the password is the remote console password.

### Q: How do I restart Node-RED from the Venus GUI?

In order to restart Node-RED you need to disable and enable it from the menu. You can find this under *Venus OS (Operating System) Large Features* → *Node-RED*.

### Q: How does safe mode work and what is it?

From the Venus GUI (Graphical User Interface) you can select Node-RED to be enabled in "safe mode". This means that Node-RED will start without starting the flows. This allows you to open the flows in the editor and make changes without the flows running. When you deploy your changes, the flows are then started. The safe mode can be handy if one of the flows has an error that causes high CPU loads on the Venus device or other issues.

## Q: Can I install the @signalk/node-red-embedded?

No, at least not into the node-red instance which comes as part of Venus OS (Operating System) Large. To @signalk/node-red-embedded, we recommend to use the node-red plugin/instance that can optionally be installed into signalk:

1. enable Node-RED in the Signalk server, using its dashboard.
2. disable the Venus OS (Operating System) Node-RED, in GX Device → Settings → Venus OS (Operating System) Large → Node-RED
3. install the Victron pallette into the Node-RED thats part of Signalk server

## Q: I get the error: Failed to remove, unrecognised module

This is seen when trying to remove a module that has been previously uninstalled, but left some files behind. After which Node-RED is incapable of completely removing the module. This is probably a remnant of upgrading an old version of Node-RED to a newer. What we've seen is that the module is still on the disk, but Node-RED no longer is aware of it in its configuration. The way to recover from this is to add the module to the Node-RED configuration again. This can be done by performing the following steps:

1. Determine the name and version of the complaining package. This information can be found in the pallette manager.
2. Stop Node-RED (can be done via the remote console)
3. Create a backup of your package.json file in /data/home/nodered/.nodered/.
4. Edit the /data/home/nodered/.node-red/package.json file
5. Add the complaining module and version to the dependencies part of the package.json file. Make sure that the file remains valid JSON. Comma's and quotes are easily overlooked.
6. Start Node-RED

After following these steps Node-RED should know about the module again and you should be able to remove or upgrade it.

## Q: The browser complains about non private connection

Node-RED on Venus now uses SSL for its connection. As the certificate is signed by a "self signed certificate", your browser is unable to trust the connection and throw a warning ("Your connection is not private" or something similar). In order to proceed, you need to press the "Advanced" or "Show Details" button and then click to visit the website. Some browsers will still display "Not secure" and claim that the Certificate is invalid, but that is inherent to self-signed certificates. The benefit is that usernames and passwords no longer travel in clear text over the network.

Another implication might be that you need to prepend the connection with https instead of http and change the port from 1880 to 1881. So you should connect to <https://venus.local:1881/> [https://venus.local:1881/] for Node-RED. If your browser still wants to connect to the old http way, you may get an `ERR_EMPTY_RESPONSE`. Adjust the url in your browser to connect to the https site instead.

Finally, if you want to use the Node-RED command line interface for administration, you will run into the `Error: self signed certificate` error. This can be solved by setting the `NODE_EXTRA_CA_CERTS` environment variable to `/data/etc/ssl/venus.local.crt` like so:

```
export NODE_EXTRA_CA_CERTS=/data/etc/ssl/venus.local.crt
```

## 6. Signal K

### 6.1 Introduction

#### 6.1.1 Why have Signal K on your GX Device?

By adding Signal K, the GX device becomes a data hub for all marine electronics on your yacht. For example audio, boat speed and wind data, and more. The Signal K is the interface between all those devices and protocols on one side, and mobile Apps on phones, tablets and the alikes on the other. Note that SignalK itself is not a visualisation tool, its purpose is purely to gather all the data and make it available for web- phone-, and tablet-apps for the visualisation.

Also, there are SignalK plugins to forward the data to influxdb & Grafana.

Popular Apps compatible with Signal K:

- Wilhelm SK <https://www.wilhelmsk.com/> - WilhelmSK is a highly customizable boat instrument display available for iPhone, iPad, Apple Watch and AppleTV.
- ... (welcome to add more here)

#### 6.1.2 Instruments compatible with Signal K

- All Victron products that can be connected to a GX device
- NMEA0183 and NMEA2000 GPS data, AIS data, Wind Temperature and other environmental data, tbd
- Engine data available on NMEA2000 and SAE 1939
- Fusion Stereo

And many more.

#### 6.1.3 What is Signal K ?

Signalk-server is a software application that, in the context of this manual, runs on your GX Device and interface to all Victron protocols and more. It uses plugins, and there is a large community of enthusiasts around it, making it one of the most flexible and complete pieces of software that understands the plethora of protocols of all different manufacturers that make marine electronics.

For more information on Signal K, read [the Signal K website \(http://signalk.org/overview.html\)](http://signalk.org/overview.html). And our blog post of a while ago.

Note that Signal K is not a Victron project. At Victron, we are great fans of sailing, open source as well as community driven projects. And Signal K is a perfect example of such project. And when running on a GX Device its a perfect marriage.

#### 6.1.4 Signal K - the data format

Strictly speaking, Signal K is not software but a data format. And Signalk-server is the name of the software.

This section explains the rationale behind the data format.

Signal K is a modern and open data format for marine use. Its goal is to

- create a shared data model for marine related data - make marine related data easy to access, handle and build applications on

The shared data model allows you to handle the data in a uniform way independent of how the data is produced. In practice this means that you can refer to your battery voltage with `'electrical/batteries/1/voltage'` and not care where data came from: Victron BMV, NMEA 2000 bus or some other voltage sensor.

Easy access means that you can use modern standards like HTTP, WebSockets and JSON that are in widespread. There is a wealth of knowledge and skilled people are easy to find, with no special marine related knowledge needed to get started.

In Venus OS (Operating System) it brings all the data available in the GX system, including NMEA 2000 data, to a single data model and programming interface.

#### 6.1.3 Considerations on how to install Signal K

The type of install described in this document, running Venus OS (Operating System) including Signalk, has these advantages: TODO elaborate. Doing this was born out of a wish to make Signalk more mainstream, less of a raspberry / hobbyists/tinkers product.

The other common type of Signalk install, installed on a RaspberryPi running Raspbian, has the advantages of more flexibility, and ease of installing extra components such as Grafana or Influx. The disadvantage is that you are basically setting up a Linux computer yourself, which you'll also have to maintain yourself. Whereas the Venus OS (Operating System) type is more of a complete product; no need to bother about keeping individual components up to date and worrying about their compatibility.

There is no one right way for everyone to do this; the best solution will depend on the requirement.

For raspberrypi owners, note that its very easy to try both methods and change between them: simply have two sdcards.

## 6.2 Where to find support?

As above already introduced, use the Modifications space on Victron Community [https://community.victronenergy.com/search.html?c=&include=children=&f=&type=question+OR+idea+OR+kentry+OR+answer+OR+topic+OR+user&redirect=search%2Fsearch&sort=relevance&q=signal+k]. As an extra, for Signal K we recommend the Victron channel in the SignalK Slack [http://slack-invite-signalk.org/].

Do not contact normal Victron support for any help on SignalK. Its based on community support, not official support.

## 6.4 First use & configuring

### Step 1. Opening the dashboard

Open a web browser, and go to the SignalK Dashboard by navigating to <http://venus.local:3000> [http://venus.local:3000].

Further documentation about that is ... (some link to signalk-node documentation?)

### Step 2. Configure serial connections

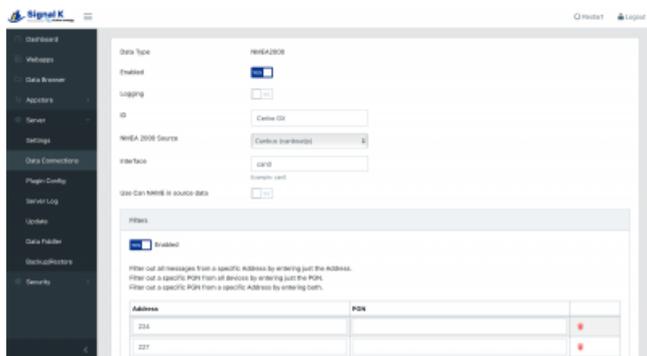
Use this to, for example, configure a USB-connected NMEA0183 connection in SignalK. To do so, go to the dashboard (<http://0.0.0.0:3000>) [http://0.0.0.0:3000], and there to Server → Connections.

Make sure to not add a connection using any of the built-in serial ports (/dev/ttyO0, /dev/ttyO2, /dev/ttyO4, /dev/ttyO5). These are the VE.Bus- and VE.Direct ports on the Venus GX, and it is not necessary to configure them in SignalK. Data from any devices there connected is already available in SignalK, by means of the "Victron Venus Plugin"; see Dashboard → Server → Plugin Config.

Note that once configured in SignalK, for example a USB-serial port, that port is no longer normally available in Venus OS (Operating System). For example, when plugging in a USB-GPS; and then adding a connection for its corresponding tty in the SignalK, makes for that GPS to be no longer be recognised and available in Venus OS (Operating System).

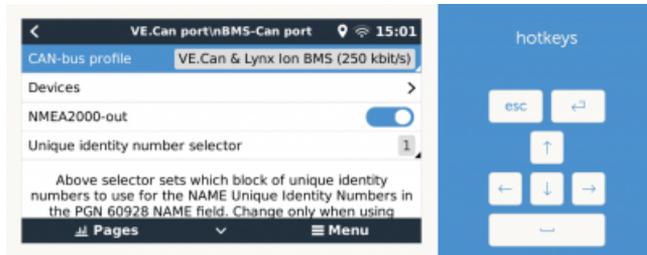
### Step 3. Configure can-bus connections

For the Venus GX, SignalK comes pre-configured with a NMEA2000 listener for the VE.Can ports, its named "n2k-on-ve-can-socket", and uses the canboatjs.



(/live/\_detail/venus-os:cerbo-gx-signalk-canboatjs.png?id=venus-os%3Alarge)

Note that, to keep that operational, that port must be configured as a 250kbit VE.Can & NMEA2000 port in the Remote Console.



(/live/\_detail/venus-os:cerbo-gx-vecan-nmea2000.png?id=venus-os%3Alarge)

### Step 4. Adding Signalk Plugins or Webapps

TODD: explain & warn about data partition size & space (since thats where plugins are installed by signalk on Venus OS (Operating System)). Filling up the datapartition can corrupt Venus OS (Operating System). More information about data partition on Venus OS (Operating System) here: <https://github.com/victronenergy/venus/wiki/data-partition> [https://github.com/victronenergy/venus/wiki/data-partition]

## 6.5 Signal K FAQ

### Q1 How to best handle & install a GPS

You want the data available in SignalK and in Venus OS (Operating System), so that it gets to VRM. And preferably a quality GPS.

#### USB GPS Dongle

Plugging it into the GX Device will make it be recognised automatically; within Venus OS (Operating System) itself as well as in SignalK, using the victron venus plugin. No need to configure anything.

#### NMEA2000 GPS

Connect it to the GX Device on the canbus port. Consider using our VE.Can to NMEA2000 cable to go from the NMEA2000 Micro-C cables to the RJ-45 on the GX Device.

In Venus OS (Operating System) v2.60, support for NMEA2000 GPS-es was added.

Signal K can then get it in two ways: directly from the NMEA2000 canbus in case thats configured as a data provider in Signal K. And via the native Venus OS (Operating System) functionality through the Victron-signalk-plugin.

Conclusion (untested), when using a NMEA2000 GPS, it will be available to both Venus OS (Operating System) and Signal K.

#### Part of a NMEA0183 network

Tbd. Actually a NMEA0183-network is probably something that doesnt exist: you can't have NMEA0183 in a network, its point to point.

## 7. Remote access via VRM

The three available web interfaces, the Node-RED flow editor, Node-RED dashboard and the Signal K admin panel can also be accessed remotely, via VRM.

After enabling them, to the VRM, and there click the Venus OS (Operating System) Large menu. Note that this might be slow, depending on the quality of the connection.

my Venus OS Large install - VRM x +

vrm.victronenergy.com/installation/71749/proxy

**victron energy**  
BLUE POWER

Search an installation

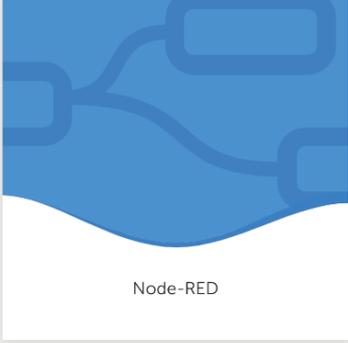
< BACK

- Dashboard
- Advanced
- Device list
- Venus OS Large**
- Settings +
- Remote Console
- Notes
- Photos
- Share
- Alarm logs
- Reporting +
- Diagnostics

Log out

## Venus OS Large

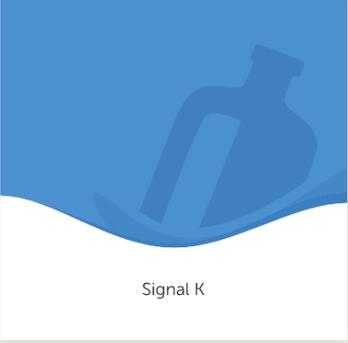
Click on a service to proxy and open it in your browser. Read more about [Node - RED](#) and [Signal K](#).



Node-RED



Node-RED Dashboard



Signal K