MeshCom App

User Manual

for iOS and Android

Version: App 4.30, firmware 4.40 a

October 2026

The MeshCom app controls MeshCom nodes over Bluetooth: messages, configuration, sensor values and map.
App author: OE1KFR, Rainer.
https://icssw.org/en/meshcom-2-0-protokoll/ · https://github.com/icssw-org/MeshCom-Firmware

Contents

1. Overview
- 1.1 The six tabs
- 1.2 Conventions

2. Installation and permissions
- 2.1 Permissions

3. Devices tab: connecting
- 3.1 The header card
- 3.2 The node cards
- 3.3 Connecting and disconnecting
- 3.4 BLE PIN per node
- 3.5 Auto reconnect
- 3.6 Monitoring the loading of the configuration
- 3.7 Connection lost, error messages

4. Settings tab: configuring the node
- 4.1 Header and display cards
- 4.2 Callsign
- 4.3 WiFi Settings
- 4.4 APRS Settings
- 4.5 Custom BLE PIN 6 Digits
- 4.6 Set Location - Phone GPS and Manual Location
- 4.7 Node Time Zone
- 4.8 Country Setting
- 4.9 Group Subscription
- 4.10 User Buttons
- 4.11 TX-Power
- 4.12 Gateway Server
- 4.13 Max Hop (only with new firmware)
- 4.14 Hardware Pins
- 4.15 Sensor Settings
- 4.16 Max. Battery Voltage (only with new firmware)
- 4.17 Network: Fixed IP, IP MTU, Ext. UDP, Remote Netconsole, Manual Routing, Web password
- 4.18 Manual Command
- 4.19 Battery Display (only with new firmware)
- 4.20 Advanced Settings

5. Info tab
- 5.1 The log

6. Chat tab
- 6.1 The chat list
- 6.2 Reading messages
- 6.3 Sending messages
- 6.4 Message menu (long press)
- 6.5 Filters: blocking callsigns and hiding texts
- 6.6 Favourites for direct messages
- 6.7 Firmware update notice

7. Map tab
- 7.1 Markers
- 7.2 The buttons

8. Heard Nodes tab

9. In practice: typical procedures
9.1 Putting a new node into operation
9.2 Running a node as a gateway
9.3 Firmware update by OTA
9.4 Using a protected connection to a node

10. Troubleshooting

11. Firmware versions

Appendix A: All settings and their function at a glance

Appendix B: Important limits

Appendix C: Time zone presets

1. Overview

The MeshCom app for iOS and Android is the remote control for a MeshCom node. It connects to the node over Bluetooth Low Energy (BLE) and is used to send and receive messages, configure the node, show measured values and show the heard stations on a map. The phone needs no internet connection to work with the node; only the map tiles in the Map tab are loaded online.

This manual describes **app version 4.30** together with **firmware 4.40 a**. With older firmware the app keeps working, but some settings are missing (see chapter 11).

1.1 The six tabs

The tab bar at the bottom of the app has six icons, from left to right:

Icon	Tab (page title)	Purpose
Bluetooth	**Devices**	Find nodes in range, connect, disconnect. Auto reconnect, BLE PIN per node (chapter 3).
Gear	**Node Settings**	Configure the connected node (chapter 4).
Circle with i	**Info**	Battery, GPS, sensor values, firmware version, app log (chapter 5).
Speech bubble	**Chat**	Read and write messages (chapter 6). A green dot on the icon shows unread messages.
Globe	**Node Map**	Map with the received nodes (chapter 7).
Arrow symbol	**Heard Nodes**	List of the stations heard directly over radio, with reception values (chapter 8).

1.2 Conventions

- **Green check button** (circle with a check mark): the entered value is sent to the node and saved there.
- **RST**: resets the setting on the node to its default (for example to delete a password or comment).
- **Collapsible sections**: most settings are folded into sections. Tap the heading to open or close a section (arrow on the right).
- **Filled and outlined buttons**: a filled button means "active/on", an outlined one "inactive/off" (exception DISPLAY, see 4.10).
- **Feedback from the node**: tapping a switch only sends the command. The button changes its state when the node has processed the command and confirmed it. This can take a few seconds.

- **Design**: the app follows the system design of the phone (light or dark).

2. Installation and permissions

The app is available in the Apple App Store and in the Google Play Store. The current download instructions are on the project page at https://icssw.org/meshcom-app/ . Firmware, flashing and OTA update are described on the project pages (https://icssw.org/meshcom-ota-update/).

2.1 Permissions

On first start the app asks for the following permissions. Without the first one, connecting is not possible.

Permission	Purpose	Platform
Bluetooth	Find and connect to nodes.	iOS: "Bluetooth". Android 12 and newer: "Nearby devices" (scan and connect).
Location	Use the position of the phone as the position of the node ("Set Location - Phone GPS"). On Android before version 12 the Bluetooth scan also needs the location permission.	iOS and Android
Notifications	Alert and sound for a new message, also when the app runs in the background.	iOS and Android

> **Note:** Messages, positions and chat settings are stored in a local database on the phone. Messages and positions are deleted automatically after 3 days.

3. Devices tab: connecting

The Devices tab is the start screen. The app scans continuously for nodes in range and shows each node found as a card.

3.1 The header card

The "Nearby Nodes" header card shows the state in its second line and carries two switches:

Display	Meaning
Scanning, tap to connect	The scan is running. Tap a card to connect.
Connecting...	The connection is being set up.
Tap the node to disconnect	Connected. Tap the card to disconnect again.
Searching for the lost node... / Signal too weak, move closer to the node	Auto reconnect is working (see 3.5).
Scan paused	The scan is stopped (for example when Bluetooth is switched off on the phone).

Switch	Effect
MeshCom only	On: only devices whose Bluetooth name starts with "MC-" are shown. Off: all BLE devices nearby appear.
Auto reconnect	On: when the connection is lost, the app searches for the node on its own and reconnects (see 3.5). The setting is saved.

3.2 The node cards

Each card shows:

- a **state icon** on the left: red cross = not connected, spinner = connecting, green check = connected;
- the **name** of the node: small, the identifier in front of the callsign (for example "MC-1c24"); large and bold, the callsign. The name is read directly from the Bluetooth advertisement and the MeshCom manufacturer data;
- the **signal strength** (RSSI in dBm) with four signal bars on the right. While connecting, "Connecting" or "Connected" is shown there;
- the **key icon** for the BLE PIN of this node (see 3.4).

Bars	RSSI	Rating
4	-70 dBm or better	very good
3	-71 to -80 dBm	good
2	-81 to -90 dBm	ok
1 (orange)	-91 to -110 dBm	weak
1 (red)	worse than -110 dBm	very weak, connecting becomes unreliable

> **Tip:** If the app does not find the node: is it switched on? Is Bluetooth on at the phone? Move closer. A node that is connected to another device sends no Bluetooth advertisement and does not appear in the list. A node without a callsign setting is called "XX0XXX-00".

3.3 Connecting and disconnecting

Tap the card of the node you want. While connecting and as long as the connection is up, the list shows only this one card. Another tap on the card disconnects; only then do the other nodes appear again.

After connecting, the following happens:

1. The app sends a "Hello" to the node (with the BLE PIN, if stored).
2. The node sends its configuration. A window "Loading Config from node. Please wait..." shows the progress.
3. The app then switches to the **Chat tab** automatically. Stored messages and the list of heard stations come from the node.
4. Every 20 seconds the app sends the time of the phone to the node (time synchronisation).

> **Note:** New or erased node: if the node has no callsign yet ("XX0XXX-00") or no position, "Unconfigured Node!" appears and the app jumps to the Settings. The app gives the node the position of the phone as start position. Setting the callsign and saving settings: see 4.2.

3.4 BLE PIN per node

Each node can be protected against unauthorised connections with a six-digit PIN. The default is **000000** (no protection). The app remembers the PIN **per node on the phone**.

The key icon on the node card opens the PIN window:

- **Current PIN**: the PIN stored on the phone for this node, or "No PIN set — open connection."
- **New PIN (6 digits)** and **Save PIN**: saves the PIN on the phone.
- **Clear PIN**: deletes the stored PIN on the phone.

> **Warning:** The PIN window only changes the entry on the phone ("Pin change only in Phone DB"), not the PIN in the node. The PIN in the node is set in the Settings (4.5, "Custom BLE PIN 6 Digits"), where it is also stored on the phone. If the stored PIN does not match the one in the node, the node drops the connection without an error message; the app then shows "BLE Disconnect ... Check the BLE Pin".

Bulk function: Settings → Advanced Settings → **Clear All BLE PINs** deletes the stored PINs of all nodes on the phone.

3.5 Auto reconnect

If the Bluetooth connection is lost (phone out of range, node restarted, radio interference), the app does not just show a message, it **searches for the node on its own and reconnects**. The prerequisite is the **Auto reconnect** switch in the Devices tab.

Sequence

1. An unexpected loss of the connection starts a cycle. At the top of the app the banner **"Auto reconnect"** appears with the progress and a **Cancel** button. It appears in every tab.

2. The cycle **scans first**. It connects only if the node was heard recently (within 10 seconds), the smoothed reception reaches at least **-90 dBm**, at least 2 advertisements of the node arrived in 8 seconds, it has been heard for at least 3 seconds and two measurements in a row were good. This prevents connections that would drop again immediately.
3. When the conditions are met, the app connects silently in the background (up to 3 attempts with growing waiting time). After connecting it loads the configuration as for a normal connection.
4. The cycle searches for 60 seconds at most. If the node is not found or cannot be connected by then, it ends with a message.

Messages in the banner

Text	Meaning
Searching for the node... It is not in range yet.	The node cannot be heard yet.
Node found (-xx dBm). Checking that the signal is stable...	Node heard, the app briefly checks that the signal is stable.
Link quality too low (-xx dBm). The node was heard only n times in 8 s, 2 needed. Move closer to the node.	The signal strength is fine, but the node is heard only rarely. Move closer. No connection is tried until then.
Signal too weak (-xx dBm, -90 dBm or better needed). Move closer to the node.	Reception too weak. Move closer.
Reconnecting... (attempt n/3)	The connection is being set up.

When the cycle ends without a connection

The app then shows the "BLE Disconnect" notice with the reason:

Message	Reason / what to do
Node not in range. Move closer and reconnect manually.	The node was not heard within 60 seconds.
Signal too weak ... The reconnect was not tried	Reception below -90 dBm.
Link quality too low ...: the node was heard only rarely.	The node is heard only rarely.
The connection keeps dropping. Check the signal strength of the node.	The connection dropped three times within two minutes shortly after reconnecting. The app gives up so as not to keep bouncing.
Reconnect failed: the connection could not be established ...	All connection attempts failed, usually because of poor link quality.

When there is no auto reconnect

- The connection was **disconnected by hand** (tap on the connected card).
- The node was sent to **Deep Sleep** or into **OTA update mode** through the app (the OTA firmware has no Bluetooth). The disconnect is then expected and the app shows only the normal disconnect message.
- The **Auto reconnect** switch is off.
- A tap on a card or a tap on **Cancel** ends a running cycle at once.

App in the background and app restart

- **Loss in the background**: scan and app logic are not reliable in the background. The cycle starts as soon as the app is in the foreground again.
- **Cold start**: if the operating system ends the app in the background (iOS after hours, Android depending on the device after about one hour with a locked phone) and the connection was up without having been disconnected by hand, the app reconnects the **last connected node** at the next start. For this it switches to the Devices tab.

3.6 Monitoring the loading of the configuration

After connecting, the app monitors whether all configuration data arrives. If no data arrives for 10 seconds, it requests the missing parts again automatically. A **"Loading Config"** window shows the progress (for example "7 / 14 received") and the attempt ("attempt n/m"). If nothing arrives at all, the app sends the Hello once more. **Cancel** stops the reloading. At the end it says "Config loaded completely."

3.7 Connection lost, error messages

Message	Cause / remedy
BLE Disconnect – Node disconnected! Check the BLE Pin and reconnect to Node!	Connection dropped or wrong PIN. Check the PIN (3.4), reconnect.
Error Pairing! Remove Node from BLE Devices after Erase/Flash	The phone still has an old pairing entry, for example after flashing again. In the Bluetooth settings of the phone remove ("forget") the node and connect again.
Error Connection! Remove Node from BLE Devices after Erase/Flash	Timeout with a node in factory state. Same remedy.
Error on Setting up connection! Remove node from BLE Devices and pair newly!	The Hello could not be sent. Remove the node on the phone and pair again.
Error on Connecting to Node! Please reconnect to Node!	The connection was made, but the notifications could not be started. Connect again.

> **Tip:** Android: error code 133 ("GATT_ERROR") is a generic error when setting up the connection. For a manual connect the app therefore retries once on its own. If that does not help, switch Bluetooth off and on at the phone briefly and connect again.

4. Settings tab: configuring the node

The "Node Settings" tab shows the settings of the connected node and lets you change them. It only makes sense when a node is connected. This chapter follows the order in which the settings appear in the app. For each setting the matching firmware command is given, so you can find it in the firmware documentation or send it with **Manual Command** (4.18).

> **Note:** Settings marked "only with new firmware" appear only when the node reports them (see chapter 11).

4.1 Header and display cards

At the very top are the **callsign** of the node, the transmit frequency (**QRG**, in MHz) and the transmit power (**TX Pwr**, in dBm). Below follow read-only cards:

- **APRS**: Symbol ID, Symbol, Comment and Name (Comment and Name only if set).
- **Call Groups**: the subscribed groups (occupied slots only).
- **WiFi** (or **Ethernet** on the RAK4631): SSID, IP, Gateway, Subnet Mask, DNS and, if set, NTP of the node.

4.2 Callsign

Input field "Set Node Callsign", for example `OE1KFR-1`. The app converts to upper case, removes a leading zero of the SSID (`OE1KFR-02` becomes `OE1KFR-2`) and checks the format (callsign, hyphen, SSID 0–99, at most 11 characters). The check mark sends it.

> **Warning:** After saving, the node reboots after about 15 seconds. The Bluetooth connection drops. Connect again afterwards. The Devices list may still show the old name until the node is seen again.

Firmware command: `--setcall`.

4.3 WiFi Settings

SSID (at most 33 characters) and password (at most 64 characters) of the WLAN the node should join. The eye icon shows or hides the password. Check mark: save; the node reboots after about 15 seconds. **RST** deletes SSID and password ("Wifi Settings reset to none").

> **Note:** The node builds a WLAN connection only when it runs as a gateway or the web server is enabled. The RAK4631 has an Ethernet interface instead of WLAN; this section has no function there.

Firmware commands: `--setssid`, `--setpwd`.

4.4 APRS Settings

Collapsible section with three cards that determine how the node appears in the map, in the dashboard and on aprs.fi.

Card	Function	Command
APRS Symbol	Map symbol. Either a preset from the "APRS Map Symbol Preset" list or by hand: "Group Character" (symbol table) and "Character" (symbol code), one character each.	`--symid`, `--symcd`
APRS Comment	Short description of the node, at most 15 characters. RST deletes the comment.	`--aprscomment` (`--atxt`)
APRS Name	First name of the operator, at most 15 characters. RST deletes the name.	`--setname`

4.5 Custom BLE PIN 6 Digits

Sets the six-digit Bluetooth PIN of the node (digits only, 100000 to 999999). Check mark: set the PIN in the node **and** store it on the phone. **RST** resets the PIN in the node to `000000` and deletes the entry on the phone. How it works: see 3.4.

Firmware command: `--btcode`.

4.6 Set Location - Phone GPS and Manual Location

- **Set Location - Phone GPS** (button): the app reads the GPS position of the phone and sets it as the fixed position of the node. Meant for nodes without their own GPS.
- **Manual Location** (collapsible): type latitude, longitude and altitude (in metres) by hand. Check mark: send.

Firmware commands: `--setlat`, `--setlon`, `--setalt`. If the node has an enabled GPS module, it provides the position.

4.7 Node Time Zone

Here you set how the node converts from UTC to local time. The node shows this time on its display.

Field	Meaning
Time Zone Preset	Selection list of time zones, for example "Europe - Vienna" (also valid for Berlin, Zurich, Rome), London, Athens, Moscow, New York, Tokyo, Sydney (complete list in appendix C). If you pick a preset, the app fills in the rule.
Set Time Zone	Own POSIX rule, for example `CET-1CEST,M3.5.0,M10.5.0/3`. **With it the node switches between summer and winter time automatically.** No spaces, at most 39 characters, M rules only.
Set UTC Offset	Fixed offset to UTC in hours, for example `1` or `-5.5`. This switches the automatic change off; in summer you have to change the value yourself.

Below the fields the app shows "Automatic time change active ..." (rule active) or "Fixed offset, no automatic time change." **RST** sets the offset to 0 and removes the rule.

> **Tip:** For Austria, Germany and Switzerland: choose the preset "Europe - Vienna" and tap the check mark. Afterwards you never have to change the time again.

Firmware commands: `--settz`, `--utcoff`. A fixed offset deletes the rule. With older firmware there is only the fixed offset.

4.8 Country Setting

The country selection determines the radio parameters of the LoRa module (frequency, bandwidth, spreading factor ...) according to local rules. The frequency is shown in the list next to the code. Make the selection and tap the check mark; the node applies the parameters. All nodes that should talk to each other must have the same setting.

Code	Frequency
EU8	433.175 MHz
EA	433.175 MHz
UK / UK8	439.9125 MHz
LA	433.925 MHz
PL	434.855 MHz
US	433.175 MHz
VR2	435.775 MHz
868	869.525 MHz
915	906.875 MHz

Firmware command: `--setctry`.

4.9 Group Subscription

Up to **6 group numbers** (1 to 5 digits each) that the node subscribes to. The information goes to the nearest gateway, and messages to these groups are delivered. Check mark: save. **RST** (Reset Groups) sets all groups to 0 (no group). Each occupied group gets its own chat in the Chat tab (chapter 6). See also https://icssw.org/grc-gruppen-call/.

Firmware command: `--setgrc`.

4.10 User Buttons

The buttons send switching commands to the node. **Filled = active.** The state is updated after the node has answered.

General

Button	Function	Command
GATEWAY	The node works as a gateway between the radio network and the MeshCom server (internet or HAMNET). Messages are forwarded to the server and back. Servers: see 4.12.	`--gateway on/off`
DISPLAY	Display of the node on/off. **Exception:** here "filled" means the display is on.	`--display`
RX Gain Boost	Boosted reception. ESP32 nodes with an SX1262 LoRa chip only.	`--setboostedgain`
Webserver	Web interface of the node on/off. Needs a valid WLAN SSID with password ("Wifi Settings not set!" otherwise). On the RAK4631 without this check.	`--webserver on/off`

Button	Function	Command
MESH	On: the node forwards received foreign messages over radio (relay). Off: it only listens and sends its own messages.	`--mesh on/off`
No ALL Msgs	Hides messages to "*" or "ALL" (on the node and in the phone).	`--nomsgall on/off`
BUTTON	Switches the evaluation of a connected user button on/off (pin: see 4.14). With it you scroll through the received messages on the display.	`--button on/off`
Wifi AP	The node opens its own WLAN (access point) when no router is available. Not possible while the node is a gateway ("Gateway Mode active!").	`--wifiap on/off`

GPS - Position

Button	Function	Command
GPS	Optional GPS module on/off. With GPS "OFF" the Info tab shows no satellites and no HDOP.	`--gps on/off`
TRACK	Track function (SmartBeaconing): after movement the node sends its position as a LoRa-APRS message on 433.775 MHz.	`--track on/off`
Send POS	Sends the position once. Behaviour see below. At least 10 seconds must pass between two calls.	`--sendpos`, `--sendtrack`
POS-Info	Requests the current position data from the node. It appears in the Info tab.	`--pos`

Send POS depends on the TRACK state:

- **TRACK off**: the node sends a MeshCom position.
- **TRACK on, GPS fix available**: the node sends the position by LoRa-APRS (and additionally by MeshCom if no MeshCom packet was received for a while). The app says "LoRa-APRS Position TX triggered!".
- **TRACK on, but no GPS fix**: the question "No GPS fix – Which position should be sent?" appears. Choices are "Manual LoRa-APRS POS" (stored position by LoRa-APRS) or "Manual MeshCom POS".

Utilities

Button	Function	Command
Scan I2C	The node searches the I2C bus for connected devices (sensors, display). The app does not show the result, it appears on the console of the node. The button is also in the Sensor Settings.	`--showi2c`
OTA Update	The node restarts in update mode (confirmation "Boot into OTA Mode now?"). ESP32-based nodes only; on RAK4631, T-Echo and Heltec T114 "Wifi OTA Update not supported!" appears. See below.	`--ota-update`
REBOOT	Restarts the node (with confirmation).	`--reboot now`
Deep Sleep	Sends the node into deep sleep (with confirmation). The connection is dropped; the app does **not** reconnect automatically. Wake up at the node (button or reset).	`--deepsleep`

> **Note: OTA update**: after the restart the node can be reached over the WLAN in a web browser: `CALLSIGN.local` or the IP address of the node (shown in the Settings header). Without WLAN access it opens its own access point: address `192.168.4.1` or `Meshcom-OTA.local`. There you upload the new firmware (.bin). The OTA firmware has no Bluetooth. Detailed instructions: https://icssw.org/meshcom-ota-update/

KISS Interface (only with new firmware)

Four buttons control the KISS/TCP interface of the node (port 8001) that external programs can connect to:

Button	Function	Command
KISS	KISS/TCP port on/off.	`--kiss on/off`
TX	Transmitting over KISS is allowed.	`--kiss tx on/off`
RxMeta	KISS also delivers metadata frames (RSSI, SNR ...).	`--kiss meta on/off`
Auth	KISS requires a login. Uses the password of the Remote Netconsole (4.17). The app points this out.	`--kiss auth on/off`

4.11 TX-Power

Slider for the transmit power in dBm. Below it the power is shown in mW and as a percentage of the maximum. The maximum depends on the hardware (usually 17 to 22 dBm). For E22 nodes the app shows only the power of the SX1262; the actual power depends on the amplifier behind it in the E22 (up to 30 or 33 dBm).

Firmware command: `--txpower`.

4.12 Gateway Server

Selects the MeshCom server that a node in **gateway mode** connects to. Make the selection and tap the check mark; the node restarts.

Selection	Server for
Austria-OE	Austria
Germany-DL	Germany (internet)
Germany Hamnet-HC	Germany (HAMNET)
Italy-IT	Italy
Switzerland-HB	Switzerland
Poland-SP	Poland

Firmware command: `--gateway srv`.

4.13 Max Hop (only with new firmware)

Limits over how many stations (hops) a **text message** from your own node is passed on. Selection 2, 3 or 4; tap the check mark to save. A smaller value relieves the network.

Firmware command: `--maxhop`.

4.14 Hardware Pins

- **Userbutton Pin**: GPIO number for connecting a button (see BUTTON in 4.10).
- **OneWire Pin**: GPIO number for the optional OneWire temperature sensor. The value appears in the Info tab as "TOUT".

Firmware commands: `--button gpio`, `--onewire gpio`.

4.15 Sensor Settings

Switch sensors on the I2C bus or OneWire on and off. Filled = active. If a sensor is switched on but was not found on the bus, its button turns **red**.

Button	Sensor	Command
BME280 / BMP280 / BMP390	Temperature, air pressure (BME280 also humidity)	`--bme`, `--bmp`, `--390`
BME680	Temperature, humidity, pressure, gas resistance	`--680`
AHT-20 / SHT21	Temperature and humidity	`--aht20`, `--sht21`
MCU-811	Gas sensor (eCO2)	`--811`
One Wire	OneWire temperature sensor (pin: see 4.14)	`--onewire`
LPS33	Weather data board on the RAK4631	`--lps33`
Sensor Info	Requests the sensor values, they appear in the Info tab	`--wx`
Scan I2C	See 4.10 (Utilities)	`--showi2c`

Temperature Offset

Correction value in degrees for the temperature of the I2C sensors (**BME/BMP Offset**) and of the OneWire sensor (**Onewire Offset**), for example `-1.5` if the node reads too warm because of its own waste heat. Command: `--tempoff`.

INA226 (only if the node reports the function)

The INA226 measures voltage and current, for example at the solar installation of a site. The section shows whether the chip was found on the I2C bus ("INA226 found ... it is active") – the node detects it at start-up. You can set:

Field	Meaning	Range
Shunt	Resistance of the measuring resistor in ohm	more than 0.001 and less than 0.5
Max. current	Highest current to be measured in ampere	0 to 20
Averaged samples	Averaging. The selection 0 to 7 corresponds to 1, 4, 16, 64, 128, 256, 512 or 1024 measurements	0–7

Commands: `--shunt`, `--imax`, `--isamp`. The values appear in the Info tab (VBUS, VSHUNT, VAMP, VPOW) and in the map in the detail window ("Current").

4.16 Max. Battery Voltage (only with new firmware)

Voltage at which the battery counts as 100 % charged (3.7 to 4.5 V). The section appears only for nodes with a single-cell battery (LiIon). The node uses the value to calculate the percentage display. The app also accepts a decimal comma. Command: `--maxv`.

4.17 Network: Fixed IP, IP MTU, Ext. UDP, Remote Netconsole, Manual Routing, Web password

Fixed IP Settings

Fixed IP address instead of DHCP: IP Address, Subnet Mask, Gateway, DNS and optionally NTP (leave empty if not needed). Check mark: save, reboot after about 15 seconds. **RST** switches back to DHCP. Commands: `--setownip`, `--setowngw`, `--setownms`, `--setowndns`, `--setownntp`.

IP MTU (only with new firmware)

Size of the IP packets of the node. Selection: **1280** (HAMNET/VPN, default), **1400**, **1500** (plain LAN). With problems over HAMNET or VPN stay with the default. Command: `--mtu`.

Ext. UDP Interface

The node passes the MeshCom data on by UDP to a device in the network. Entry **UDP Dest. Addr.** (IP address of the target), check mark to save, and the **Enable** switch turns the function on or off. Commands: `--extudpip`, `--extudp on/off`.

Remote Netconsole (only with new firmware)

The console of the node can be reached over the network on **port 2323** (for example with a terminal program). **Enable** switch and a **Password** (at most 14 characters). The app does not show the password; you can only set it or delete it with **RST**. Without a password the console is open to everyone in the network. The KISS login uses the same password (see 4.10). Commands: `--netconsole`, `--passwd`.

Manual Routing (VIA)

Lets messages travel through a specific node (via callsign). Enter the **Destination Call** (for example `OE1KFR-12`), tap the check mark and set the **Enable** switch. **RST** deletes the target. With older firmware "Manual Route Settings not provided by node. Check FW Update or Re-Connect" appears. Command: `--via`.

Web UI Password

Password for the web interface of the node (at most 19 characters). The app shows the current password in plain text. **RST** removes it. Command: `--webpwd`.

4.18 Manual Command

Any node command by hand, for example `--info`. It **must start with two hyphens**. If the iOS keyboard replaces the two hyphens with a long dash, the app reports it: "It starts with a long dash instead of two hyphens". Length at most 250 bytes. Check mark: send. The list of commands is given by the command `--help` and the firmware documentation.

> **Warning:** A command takes effect immediately and without confirmation. Only send commands whose effect you know (for example `--cleanflash` deletes the settings of the node).

4.19 Battery Display (only with new firmware)

Determines whether the display of the node shows the battery as **Voltage** or **Percent**. Commands: `--volt on`, `--proz on`.

4.20 Advanced Settings

Four red buttons delete data **on the phone**. There is no confirmation.

Button	Deletes
Clear received nodes	All received node positions in the map.
Clear Text Msgs	All messages in the Chat tab.
Clear Mheards	All entries in the Heard Nodes tab.
Clear All BLE PINs	The stored BLE PINs of all nodes (see 3.4).

5. Info tab

The Info tab shows the state of the connected node. As long as the tab is visible, the app polls the node in turn for new values (one request every 4 seconds). Polling pauses when the app is in the background, no node is connected or the log window is open.

Card	Content
Battery	State of charge in percent with a bar (orange below 30 %, red below 15 %), voltage in volts and the hardware of the node (for example "HELTEC V3", "RAK4631").
GPS	Top right the state: **Fix**, **No Fix** or **OFF** (GPS switched off; satellites and HDOP are hidden then). Also Latitude, Longitude, Altitude, Satellites, HDOP, **Next POS** (time until the next position is sent, for example "5m 20s"), UTC Offset, and Date and Time in two rows.
Sensors	Measured values of the enabled sensors: Temp, TOUT (OneWire), Humidity, Press, QNH, GasRes, eCO2, Alt Press and, with the INA226, VBUS, VSHUNT, VAMP, VPOW. **Values the node does not deliver are hidden.** Below, chips show which sensors are active (BME280, BMP280, BMP390, BME680, MCU811, LPS, OneWire, AHT-20, INA226), and the OneWire Pin.
Firmware	**Version** (firmware of the node), **Build** (date of the firmware build) and the **App Version**.

5.1 The log

The **Log** button at the end of the page opens the internal log of the app. It helps with troubleshooting (connection set-up, auto reconnect, scan diagnostics). In the window:

- **Clear** (top left) deletes the log.
- **Copy** copies the whole log to the clipboard, oldest entry first, with a header line of app version, platform and time. This way it can be pasted into an e-mail.
- **Close** closes the window.

> **Tip:** With connection problems: open the log, tap **Copy** and paste the log into your support request or bug report. It contains the phases of the connection set-up with times and the scan count of the nodes.

6. Chat tab

6.1 The chat list

The Chat tab starts with a **list of chats**:

- **To All Channel**: messages to everyone (broadcast, "*" or "ALL").
- **Direct Messages**: personal messages to a callsign.
- **Group n**: one chat for each group the node subscribes to (see 4.9). It appears only if the group is occupied.

Each entry shows the last message as a preview ("CALLSIGN: text"), and for unread messages a green envelope and a counter badge. Two icons are on the right:

- **Funnel**: opens the filters of this chat (6.5). If a filter is active, the funnel is orange.
- **Speaker**: switches sound and notification sound for this chat on or off (crossed out = silent). A notification without sound still arrives.

Tapping the entry opens the chat. At the top left the **arrow back** leads to the list. The back gesture works too: on **iOS** swipe from the left edge, on **Android** use the back button.

6.2 Reading messages

A message shows the time and, if it was relayed by other stations, "· via CALLSIGN", below it the sender and the text. The app recognises links in the text and makes them tappable. When the date changes, a date bar appears. When you open a chat with unread messages, the marker **"New Messages"** stands at the first unread message. An arrow at the bottom right jumps to the end of the chat.

The app announces new messages by notification ("New Message from CALLSIGN"), also in the background. Own messages, commands that begin with "--" and filtered messages (6.5) do not trigger a notification.

6.3 Sending messages

1. Open the chat. In the **To All Channel** and in a **Group** chat it is enough to type the text.
2. In the **Direct Messages** chat the **To Callsign** field appears above the text field. Enter the callsign **with SSID** (for example `OE1KFR-1`). The star next to it opens the favourites (6.6).
3. Type the text in the **Type Message** field (at most 150 characters). The field grows up to six lines.
4. Tap the arrow at the bottom right. The text field is emptied and the keyboard disappears; the message is at the end of the chat.

> **Note:** If no node is connected, the fields are locked and the send arrow is red. Tapping it shows the message "BLE Disconnect".

Status icons of your own messages

Icon	Meaning
Check mark	**Sent**: the message has arrived at your own node.
Cloud	**Heard**: the message was heard by another node and relayed. The callsign of that station is shown next to it.

Icon	Meaning
Cloud with check mark	**Acked**: for a direct message the **recipient** has received the message (the confirmation came back). For broadcasts and groups, reception by a gateway confirms the forwarding to the MeshCom server.
Crossed-out cloud (orange)	**Unconfirmed**: a direct message stayed without confirmation for 6 minutes. The node gave up after four transmissions. If the confirmation still arrives, the icon becomes "Acked".

Direct messages are sent up to four times by the node in case they get lost on the way. The app still shows only **one** message; it recognises copies received twice and hides them.

6.4 Message menu (long press)

A long press on a message opens a menu:

Entry	Effect
Copy Text	Copies the message text.
Resend Message	Sends one of your own messages again (own messages only).
Direct Message	Switches to the DM chat with the sender as recipient (not in the DM chat itself).
Reply To	Sets the sender as recipient and puts "@CALLSIGN " in front in the text field.
Filter Text	Creates a text filter with this message text (6.5).
Add Favourite	Adds the callsign to the favourites (DM chat only, only if not yet present).
Block Callsign	Hides all messages of this callsign (6.5). Foreign messages only.

6.5 Filters: blocking callsigns and hiding texts

The funnel in the chat list opens the **Filters** window. It has two parts: **Filter Global** (applies in all chats) and **Filter** of the selected chat (applies only there). Both have:

- **Blocked Callsigns**: blocked callsigns. With the plus icon you add one (Apply), with the red cross you lift the block. **Delete All Callsign Blocks** deletes the whole list.
- **Text Filters**: messages whose text matches the filter are hidden. For each filter you choose the **scope** ("Globally" or this chat) and the **type**: **Exact Match** (the whole text is identical) or **Consists Of** (the text occurs in the message). Filters can be edited (pencil) and deleted. **Delete All Textfilters** deletes all.

When blocking through the message menu, the app asks whether **Globally** or **This Chat Only**. Filtered messages stay stored, they are just not shown.

6.6 Favourites for direct messages

The **star** next to "To Callsign" opens the **Favourites** window. With **Add** you enter a callsign, a tap on an entry takes it over as recipient, the red cross deletes it. You can also take over callsigns from the message menu (**Add Favourite**). The list stays saved.

6.7 Firmware update notice

If the node runs a firmware **older than 4.40 a**, the Chat tab shows the notice "Firmware update recommended" once after connecting. With older firmware some functions can be limited (chapter 11).

7. Map tab

The map (OpenStreetMap) shows all nodes whose position the app has received. The map tiles are loaded from the internet; without internet the background stays empty, but the markers are there. Older positions are deleted after three days.

7.1 Markers

Colour	Meaning
Purple	Your own (connected) node.
Green	Nodes your own node has heard **directly over radio** (see Heard Nodes).
Light blue	Austrian club stations and relay sites (callsigns of the form OE1XAB-1) that were not heard directly.
Blue	All other nodes, heard only through other nodes (mesh) or gateways.

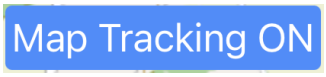
Tapping a marker opens a card with the details: callsign and APRS symbol, coordinates, altitude, battery, hardware, number of neighbours, subscribed groups, comment and name. **More** shows the sensor values, as far as the node sends them (pressure, temperature, humidity, QNH, eCO2, gas resistance, VBus and current of the INA226, the inputs of an MCP23017). **DM** switches to the chat and enters the callsign as recipient, **Close** closes the card.

7.2 The buttons

The button at the bottom right opens a menu with four functions:

	**Open/close the menu** (arrow up).
	**Lines**: shows red lines from your own node to the directly heard nodes.
	**Search**: search bar and list of all known nodes. While you type, the list filters by the beginning of the callsign. Tapping an entry centres the map on this node and opens its card.
	**Centre**: moves the map to your own node.
	**Map Tracking** on/off. The map then follows the position of the node. The node needs an enabled GPS for this. The app requests the position every 10 seconds.

At the top right the state is shown:



Zoom with **+** and **-**, or with two fingers. Map Tracking switches on by itself when the map is opened if the node has a GPS, TRACK is switched off and a connection exists.

8. Heard Nodes tab

The tab shows all stations that the **connected node has heard directly over radio**. Each station has a card with the callsign, date and time of reception and the following values. **Values the node does not send are hidden.**

Value	Meaning
RSSI (dBm)	Reception strength. Good up to -90, ok up to -100, weak up to -115, below that bad.
SNR (dB)	Signal-to-noise ratio. Very good from -5, good from -10, ok from -15, below that bad.
Dist (km)	Distance to the station, if its position is known.
Hardware	Hardware of the heard station.
Neighbours	Number of neighbours this station reports.
Hears me (dB)	How the station hears **your own node** (only with new firmware). This shows whether the radio link works in both directions.
Gateway	Appears with "Yes" if the station is a gateway.

The values are coloured: green good, yellow/orange medium, red bad. The thresholds are the ones usual for LoRa. The signal bars in the Devices tab, in contrast, measure the Bluetooth signal to the phone and have their own thresholds (3.2). The list is loaded from the node when connecting and updated during operation. To delete: Settings → Advanced Settings → **Clear Mheards**.

9. In practice: typical procedures

9.1 Putting a new node into operation

1. Switch the node on, open the Devices tab in the app and tap the card of the node.
2. Confirm the message "Unconfigured Node!". The app switches to the Settings.
3. Enter the **Callsign** and save it with the check mark. The node reboots, then connect again.
4. Choose the **Country Setting**, save.
5. Choose the **Node Time Zone** (preset), save.
6. Set the position: **Set Location - Phone GPS** or **Manual Location**; for nodes with GPS switch the GPS on (User Buttons).
7. Optional: APRS symbol and comment, groups, BLE PIN. At the first time set **TX-Power** to a sensible value.
8. In the Devices tab switch **Auto reconnect** on.

9.2 Running a node as a gateway

1. Enter the WLAN (WiFi Settings), or connect Ethernet on the RAK4631; set a fixed IP if needed.
2. Choose the **Gateway Server** that fits your country (for HAMNET "Germany Hamnet-HC").
3. Switch the **GATEWAY** button on. The node connects to the server and forwards messages.

> **Warning:** With GATEWAY switched on, **Wifi AP** cannot be switched on.

9.3 Firmware update by OTA

1. Settings → User Buttons → **OTA Update** → "YES". The node restarts in update mode, the Bluetooth connection ends (no auto reconnect).
2. Open in the browser the address the app shows (`CALLSIGN.local`, the IP address or, in AP mode, `192.168.4.1`).
3. Upload the firmware file (.bin). After the update the node restarts; connect again in the Devices tab.

9.4 Using a protected connection to a node

1. In Settings → **Custom BLE PIN 6 Digits** set a PIN and tap the check mark (the PIN is stored on the phone at the same time).
2. On **another phone**: Devices → key icon of the node card → enter the PIN → **Save PIN**, then connect.
3. PIN forgotten: reset it with the serial console or the web interface of the node; the app can do this only after connecting.

10. Troubleshooting

Problem	Possible cause and remedy
Node does not appear in the Devices tab	Node on? Bluetooth and Bluetooth/location permission granted to the app? Move closer. Switch "MeshCom only" off and on. The node sends nothing while another device is connected.
Connecting aborts at once	Wrong PIN (3.4). Remove the old pairing entry in the Bluetooth settings of the phone. On Android switch Bluetooth off and on.
On Android the node is hard to find or to connect to	It depends on the phone. Scan closer to the node (RSSI -80 dBm or better), restart the app. On iOS this rarely happens. The **log** (chapter 5.1) shows the scan count of the node.
"Loading Config" stays for a long time	This is normal with many heard stations (up to a minute). The app requests missing data itself (3.6). If that does not help, Cancel, disconnect, connect again.
After saving settings the connection is gone	The node reboots (Callsign, WiFi, fixed IP, Gateway Server). Connect again.
Button does not change its state	The app waits for the answer of the node (a few seconds). If it does not come, check the connection. With older firmware some commands do not exist.
Sensor button is red	The sensor is switched on but was not found on the I2C bus. Check the wiring, then Scan I2C at the node.
Messages are missing in the chat	Filter active (funnel orange)? Callsign blocked? "No ALL Msgs" switch active? Messages older than three days are deleted.
Direct message shows "Unconfirmed"	No confirmation from the recipient within 6 minutes. Recipient not reachable or the path is too poor. Send again with **Resend**. Always enter the callsign with SSID.
Map stays empty	No internet for the map tiles. Markers are still shown.
No notification for new messages	Notification permission granted on the phone? Speaker icon of the chat on "on"? On Android check the battery optimisation for the app.

11. Firmware versions

The app expects **firmware 4.40 a or newer**. With older firmware:

- After connecting, the notice "Firmware update recommended" appears in the Chat tab.
- With firmware older than 4.35 v the complete handling of the repetition of direct messages is missing (a node sends each copy separately, and the app then filters it only partly).
- The following settings appear **only if the node reports them**: Max Hop, IP MTU, Battery Display, Max. Battery Voltage (single-cell battery only), Remote Netconsole, KISS Interface, INA226, the time zone rule (Node Time Zone: preset and rule), Manual Routing (VIA) and the display of the web password.
- The build date of the firmware in the Info tab ("Build") is missing on older nodes.

> **Tip:** The firmware version is in the Info tab under "Firmware → Version". Firmware files and update instructions are on the MeshCom project pages.

Appendix A: All settings and their function at a glance

Section in the Settings	Function	Firmware command
Callsign	Callsign of the node (reboot)	`--setcall`
WiFi Settings	WLAN SSID and password (reboot)	`--setssid`, `--setpwd`
APRS Settings	Symbol, comment, name	`--symid`, `--symcd`, `--aprscomment`, `--setname`
Custom BLE PIN	Bluetooth PIN of the node	`--btcode`
Set Location - Phone GPS / Manual Location	Fixed position	`--setlat`, `--setlon`, `--setalt`
Node Time Zone	Time zone, automatic daylight saving	`--settz`, `--utcoff`
Country Setting	LoRa frequency and parameters	`--setctry`
Group Subscription	Subscribe to up to 6 groups	`--setgrc`
User Buttons: GATEWAY	Gateway operation	`--gateway`
User Buttons: DISPLAY	Display on/off	`--display`
User Buttons: RX Gain Boost	Boosted reception (ESP32 with SX1262)	`--setboostedgain`
User Buttons: Webserver	Web interface	`--webserver`
User Buttons: MESH	Forwarding foreign messages	`--mesh`
User Buttons: No ALL Msgs	Hide broadcasts to "*"	`--nomsgall`
User Buttons: BUTTON	Evaluate user button	`--button`
User Buttons: Wifi AP	Own access point	`--wifiap`
User Buttons: GPS	GPS module on/off	`--gps`
User Buttons: TRACK	SmartBeaconing LoRa-APRS	`--track`
User Buttons: Send POS	Send position once	`--sendpos`, `--sendtrack`
User Buttons: POS-Info	Request position data	`--pos`
User Buttons: Scan I2C	Search I2C devices	`--showi2c`
User Buttons: OTA Update	Update mode (ESP32 only)	`--ota-update`
User Buttons: REBOOT / Deep Sleep	Restart / deep sleep	`--reboot now`, `--deepsleep`
KISS Interface	KISS/TCP port, TX, metadata, login	`--kiss ...`
TX-Power	Transmit power	`--txpower`
Gateway Server	Server of the gateway (reboot)	`--gateway srv`
Max Hop	Hop limit for text messages	`--maxhop`
Hardware Pins	Pin of the user button and of OneWire	`--button gpio`, `--onewire gpio`

Section in the Settings	Function	Firmware command
Sensor Settings	Sensors on/off, Sensor Info, Scan I2C	`--bme`, `--bmp`, `--390`, `--680`, `--811`, `--aht20`, `--sht21`, `--lps33`, `--onewire`, `--wx`
Temperature Offset	Temperature correction	`--tempoff`
INA226	Shunt, maximum current, averaging	`--shunt`, `--imax`, `--isamp`
Max. Battery Voltage	Voltage for 100 % battery	`--maxv`
Fixed IP Settings	Fixed IP, gateway, DNS, NTP (reboot)	`--setownip`, `--setowngw`, `--setownms`, `--setowndns`, `--setownntp`
IP MTU	Packet size of the IP interface	`--mtu`
Ext. UDP Interface	Pass MeshCom data on by UDP	`--extudp`, `--extudpip`
Remote Netconsole	Console in the network (port 2323), password	`--netconsole`, `--passwd`
Manual Routing (VIA)	Messages through a specific node	`--via`
Web UI Password	Password of the web interface	`--webpwd`
Manual Command	Any command (starts with "--")	any
Battery Display	Display shows voltage or percent	`--volt`, `--proz`
Advanced Settings	Delete data on the phone	(app only)

Appendix B: Important limits

Value	Limit
Message text in the chat	at most 150 characters
Callsign of the recipient (To Callsign)	at most 11 characters, with SSID
Node callsign	at most 11 characters, form `OE1KFR-1`
WLAN SSID / WLAN password	33 / 64 characters
APRS Comment / APRS Name	15 / 15 characters
Web password / Netconsole password	19 / 14 characters
BLE PIN	6 digits, 100000–999999
Time zone rule	at most 39 characters, no spaces
Groups	6 slots, 1 to 5 digits each
Max. Battery Voltage	3.7 to 4.5 V
INA226 shunt / maximum current	0.001–0.5 Ω / 0–20 A
Manual Command	at most 250 bytes
Retention of messages and positions	3 days
Send POS: minimum interval	10 seconds

Appendix C: Time zone presets

Preset	POSIX rule
Europe - Vienna (CET; also Berlin, Zurich, Rome ...)	`CET-1CEST,M3.5.0,M10.5.0/3`
Europe - London	`GMT0BST,M3.5.0/1,M10.5.0`
Europe - Dublin	`GMT0IST,M3.5.0/1,M10.5.0`
Europe - Lisbon	`WET0WEST,M3.5.0/1,M10.5.0`
Europe - Athens (EET; also Helsinki, Kyiv ...)	`EET-2EEST,M3.5.0/3,M10.5.0/4`
Europe - Moscow	`MSK-3`
Europe - Istanbul	`<+03>-3`
Europe - Reykjavik	`GMT0`
America - New York / Chicago / Denver / Los Angeles	`EST5EDT,...` / `CST6CDT,...` / `MST7MDT,...` / `PST8PDT,M3.2.0,M11.1.0`
America - Phoenix (no daylight saving)	`MST7`
America - Sao Paulo	`<-03>3`
Africa - Johannesburg	`SAST-2`
Asia - Dubai / Kolkata / Shanghai / Tokyo	`<+04>-4` / `<+0530>-5:30` / `CST-8` / `JST-9`
Australia - Sydney (also Melbourne) / Perth	`AEST-10AEDT,M10.1.0,M4.1.0/3` / `AWST-8`
New Zealand - Auckland	`NZST-12NZDT,M9.5.0,M4.1.0/3`
UTC	`UTC0`

In a POSIX rule the sign is **reversed** compared with the usual notation: `CET-1` means UTC+1.

Further information: https://icssw.org/meshcom-4-0-faq/ and https://icssw.org/meshcom-app/ .