

Meshtastic Web Client and Python CLI

Meshtastic Web Client

The Meshtastic Web Client (client.meshtastic.org) provides a browser-based interface for configuring Meshtastic nodes without installing a mobile app. It supports WebSerial (Chrome/Edge) for USB connections and WebBluetooth for BLE connections on supported platforms.

Use Cases for the Web Client

- Configuring nodes on a desktop computer without a phone
- Full configuration access on laptops where the mobile app isn't practical
- Quick configuration of multiple nodes at a deployment event
- Advanced configuration options that may appear in the web client before the mobile apps

Connecting via Web Serial

1. Open client.meshtastic.org in Chrome or Edge (not Firefox)
2. Click "New Connection" → "Serial"
3. Select your device's COM port or /dev/tty device
4. The full configuration interface loads directly in the browser

Meshtastic Python CLI

The Python CLI is the most powerful configuration interface - it provides access to every configurable parameter and supports automation scripting.

Installation

```
pip install meshtastic
```

Requires a recent Python 3 (check the current minimum in the project's `pyproject.toml` on github.com/meshtastic/python). Works on Windows, macOS, and Linux.

Essential Commands

```
# Connect to a serial node and show device info
meshtastic --info

# List all nodes in the node database
meshtastic --nodes

# Set a configuration value
meshtastic --set lora.modem_preset MEDIUM_SLOW

# Set multiple values at once
meshtastic --set device.role ROUTER --set bluetooth.enabled false

# Export full configuration to a YAML file
meshtastic --export-config > config-backup.yaml

# Restore (apply) a configuration from that YAML file
meshtastic --configure config-backup.yaml

# Send a test message
meshtastic --sendtext "Test message from CLI"

# Listen to all incoming packets
meshtastic --listen

# Connect to a node via TCP (WiFi/network)
meshtastic --host 192.168.1.100 --info

# Factory reset
meshtastic --factory-reset
```

Scripting with Python API

```
import meshtastic
import meshtastic.serial_interface

# Connect
iface = meshtastic.serial_interface.SerialInterface()

# Get device info
info = iface.localNode.localConfig
print(info)

# Send a message
iface.sendText("Hello mesh!")

# Listen for messages
from pubsub import pub
def on_receive(packet, interface):
    print(f"Received: {packet}")
pub.subscribe(on_receive, "meshtastic.receive")

iface.close()
```

Specifying a Connection

```
# Serial (auto-detect)
meshtastic --info

# Serial (specific port - Windows)
meshtastic --port COM3 --info

# Serial (specific port - Linux/Mac)
meshtastic --port /dev/ttyUSB0 --info

# TCP (WiFi connection)
meshtastic --host 192.168.1.100 --info

# BLE (by node name)
meshtastic --ble "Meshtastic_ABCD" --info
```

