



Ouster Sensor API Guide

Firmware v3.1.0-2-g61e686e for all Ouster sensors

Ouster

Apr 30, 2024

Sensor API

1	Sensor Configuration	4
1.1	Overview	4
1.2	Description	5
1.2.1	<code>udp_dest</code>	5
1.2.2	<code>udp_port_lidar</code>	6
1.2.3	<code>udp_port_imu</code>	6
1.2.4	<code>sync_pulse_in_polarity</code>	6
1.2.5	<code>sync_pulse_out_polarity</code>	6
1.2.6	<code>sync_pulse_out_frequency</code>	7
1.2.7	<code>sync_pulse_out_angle</code>	7
1.2.8	<code>sync_pulse_out_pulse_width</code>	7
1.2.9	<code>nmea_in_polarity</code>	7
1.2.10	<code>nmea_ignore_valid_char</code>	8
1.2.11	<code>nmea_baud_rate</code>	8
1.2.12	<code>nmea_leap_seconds</code>	8
1.2.13	<code>azimuth_window</code>	9
1.2.14	<code>signal_multiplier</code>	9
1.2.15	<code>udp_profile_lidar</code>	9
1.2.16	<code>udp_profile_imu</code>	10
1.2.17	<code>phase_lock_enable</code>	10
1.2.18	<code>phase_lock_offset</code>	10
1.2.19	<code>lidar_mode</code>	10
1.2.20	<code>timestamp_mode</code>	11
1.2.21	<code>multipurpose_io_mode</code>	11
1.2.22	<code>operating_mode</code>	12
1.2.23	<code>min_range_threshold_cm</code>	12
1.2.24	<code>return_order</code>	12
1.2.25	<code>imu_data_format</code>	13
2	HTTP API Reference Guide	14
2.1	Sensor Metadata	14
2.1.1	GET /api/v1/sensor/metadata/sensor_info	14
2.1.2	GET /api/v1/sensor/metadata/lidar_data_format	15
2.1.3	GET /api/v1/sensor/metadata/imu_data_format	16
2.1.4	GET /api/v1/sensor/metadata/beam_intrinsics	16
2.1.5	GET /api/v1/sensor/metadata/imu_intrinsics	17
2.1.6	GET /api/v1/sensor/metadata/lidar_intrinsics	19
2.1.7	GET /api/v1/sensor/metadata/calibration_status	19
2.1.8	GET /api/v1/sensor/config	20
2.1.9	POST /api/v1/sensor/config - Multiple configuration parameters	21
2.1.10	GET /api/v1/sensor/config/operating_mode	23
2.1.11	POST /api/v1/sensor/config/operating_mode	23
2.1.12	DELETE /api/v1/sensor/config	24
2.1.13	GET /api/v1/sensor/metadata	24
2.2	User Editable Data	26
2.2.1	GET /api/v1/user/data	27
2.2.2	PUT /api/v1/user/data	27
2.2.3	DELETE /api/v1/user/data	28

2.2.4	Optional Parameters - data policy	28
2.2.5	Optional Parameters - include_metadata	29
2.3	System	30
2.3.1	POST /api/v1/system/restart	30
2.3.2	GET /api/v1/system/firmware	31
2.3.3	POST /api/v1/system/firmware	31
2.3.4	GET /api/v1/system/network	32
2.3.5	GET /api/v1/system/network/ipv4	32
2.3.6	GET /api/v1/system/network/ipv4/override	33
2.3.7	PUT /api/v1/system/network/ipv4/override	33
2.3.8	DELETE /api/v1/system/network/ipv4/override	35
2.3.9	GET /api/v1/system/network/speed_override	36
2.3.10	PUT /api/v1/system/network/speed_override	36
2.3.11	DELETE /api/v1/system/network/speed_override	37
2.4	Time	37
2.4.1	GET /api/v1/time	37
2.4.2	GET /api/v1/time/sensor	40
2.4.3	GET /api/v1/time/system	41
2.4.4	GET /api/v1/time/ptp	42
2.4.5	GET /api/v1/time/ptp/profile	44
2.4.6	PUT /api/v1/time/ptp/profile	45
2.5	Alerts, Diagnostics and Telemetry	45
2.5.1	GET /api/v1/sensor/alerts	45
2.5.2	GET /api/v1/sensor/alerts?cursor=1	49
2.5.3	GET /api/v1/sensor/alerts?mode=summary	49
2.5.4	GET /api/v1/sensor/alerts?cursor=2&mode=summary	50
2.5.5	GET /api/v1/diagnostics/dump	51
2.5.6	GET /api/v1/sensor/telemetry	52
3	TCP API Guide (Deprecated)	53
4	API Changelog	54
4.1	Firmware v3.1.0	54
4.2	Firmware v3.0.1	54
4.3	Firmware v3.0.0	55

This on-sensor reference document is a quick guide to the TCP and HTTP APIs. For the most up-to-date and comprehensive information about your sensor, please see the sensor user manuals found at www.ouster.com

1 Sensor Configuration

1.1 Overview

Overview of sensor configuration parameters. For detailed description of each parameter refer to [Description](#).

Table1: Overview

Parameter	Type	Valid Values
<code>udp_dest</code>	String	"" (default)
<code>udp_port_lidar</code>	Integer	7502 (default)
<code>udp_port_imu</code>	Integer	7503 (default)
<code>sync_pulse_in_polarity</code>	Keyword	ACTIVE_HIGH (default) ACTIVE_LOW
<code>sync_pulse_out_polarity</code>	Keyword	ACTIVE_LOW (default) ACTIVE_HIGH
<code>sync_pulse_out_frequency</code>	Integer >= 1	1 (default)
<code>sync_pulse_out_angle</code>	Integer [0 ... 360]	360 (default)
<code>sync_pulse_out_pulse_width</code>	Integer >= 0	10 (default)
<code>nmea_in_polarity</code>	Keyword	ACTIVE_HIGH (default) ACTIVE_LOW
<code>nmea_ignore_valid_char</code>	Integer	0 (default) 1
<code>nmea_baud_rate</code>	Keyword	BAUD_9600 (default) BAUD_115200
<code>nmea_leap_seconds</code>	Integer >= 0	0 (default)
<code>azimuth_window</code>	List	[0,360000] (default)
<code>signal_multiplier</code>	Number	0.25 0.5 1 (default) 2 3
<code>udp_profile_lidar</code>	Keyword	RNG19_RFL8_SIG16_NIR16 (default) RNG19_RFL8_SIG16_NIR16_DUAL RNG15_RFL8_NIR8 FUSA_RNG15_RFL8_NIR8_DUAL
<code>udp_profile_imu</code>	Keyword	LEGACY (default)

continues on next page

Table 1 - continued from previous page

Parameter	Type	Valid Values
<code>phase_lock_enable</code>	Boolean	<code>false</code> (default) <code>true</code>
<code>phase_lock_offset</code>	Integer [0 ... 360]	0 (default)
<code>lidar_mode</code>	Keyword	512x10 1024x10 (default) 2048x10 512x20 1024x20
<code>timestamp_mode</code>	Keyword	TIME_FROM_INTERNAL_OSC (default) TIME_FROM_PTP_1588 TIME_FROM_SYNC_PULSE_IN
<code>multipurpose_io_mode</code>	Keyword	OFF (default) INPUT_NMEA_UART OUTPUT_FROM_INTERNAL_OSC OUTPUT_FROM_SYNC_PULSE_IN OUTPUT_FROM_PTP_1588 OUTPUT_FROM_ENCODER_ANGLE
<code>operating_mode</code>	Keyword	NORMAL (default) STANDBY
<code>min_range_threshold_cm</code>	Number	0 30 50 (default)
<code>return_order</code>	Keyword	STRONGEST_TO_WEAKEST (Default), NEAREST_TO_FARTHEST FARTHEST_TO_NEAREST

1.2 Description

1.2.1 `udp_dest`

Description:

- Type: String
- Default: "192.0.2.123"
- Destination to which the sensor sends UDP traffic.

Note: As of now, setting the `udp_dest` to "@auto" is only supported through `HTTP POST /api/v1/sensor/config`. Parameter `udp_ip` has been deprecated in firmware v2.4, please use the `udp_dest` parameter.

1.2.2 `udp_port_lidar`

Description:

- Type: Integer [0 ... 65535]
- Default: 7502
- The `<port>` on `udp_dest` to which lidar data will be sent (7502, default).

1.2.3 `udp_port_imu`

Description:

- Type: Integer [0 ... 65535]
- Default: 7503
- The `<port>` on `udp_dest` to which IMU data will be sent (7503, default).

1.2.4 `sync_pulse_in_polarity`

Description:

- Type: Keyword
- Default: "ACTIVE_HIGH"
- **Enum:**
 - "ACTIVE_HIGH"
 - "ACTIVE_LOW"
- The polarity of SYNC_PULSE_IN input, which controls polarity of SYNC_PULSE_IN pin when `timestamp_mode` is set in `TIME_FROM_SYNC_PULSE_IN`.

1.2.5 `sync_pulse_out_polarity`

Description:

- Type: Keyword
- Default: "ACTIVE_HIGH"
- The polarity of SYNC_PULSE_OUT output, if the sensor is set as the master sensor used for time synchronization.

1.2.6 `sync_pulse_out_frequency`

Description:

- Type: Integer ≥ 1
- Default: 1
- The output SYNC_PULSE_OUT pulse rate in Hz. Valid inputs are integers >0 Hz, but also limited by the criteria described in the Time Synchronization section of the Firmware User Manual.

1.2.7 `sync_pulse_out_angle`

Description:

- Type: Integer [0 ... 360]
- Default: 360
- The angle in terms of degrees that the sensor traverses between each SYNC_PULSE_OUT pulse. E.g. a value of 180 means a sync pulse is sent out every 180° for a total of two pulses per revolution and angular frequency of 20 Hz if the sensor is 1024x10 Hz lidar mode. Valid inputs are integers between 0 and 360 inclusive but also limited by the criteria described in the Time Synchronization section of Firmware User Manual.

1.2.8 `sync_pulse_out_pulse_width`

Description:

- Type: Integer ≥ 0
- Default: 10
- The polarity of SYNC_PULSE_OUT output, if the sensor is set as the master sensor used for time synchronization. Output SYNC_PULSE_OUT pulse width is in ms, increments in 1 ms. Valid inputs are integers greater than 0 ms, but also limited by the criteria described in the Time Synchronization section of Firmware User Manual.

1.2.9 `nmea_in_polarity`

Description:

- Type: Keyword
- Default: "ACTIVE_HIGH"
- **Enum:**
 - "ACTIVE_HIGH"
 - "ACTIVE_LOW"

- Set the polarity of NMEA UART input \$GPRMC messages. See Time Synchronization section in sensor user manual for NMEA use case. Use `ACTIVE_HIGH` if UART is active high, idle low, and start bit is after a falling edge.

1.2.10 `nmea_ignore_valid_char`

Description:

- Type: Integer [0 ... 1]
- Default: 0
- Set `0` if NMEA UART input \$GPRMC messages should be ignored if valid character is not set, and `1` if messages should be used for time syncing regardless of the valid character.

1.2.11 `nmea_baud_rate`

Description:

- Type: Keyword
- Default: "BAUD_9600"
- **Enum:**
 - "BAUD_9600"
 - "BAUD_115200"
- `BAUD_9600` (default) or `BAUD_115200` for the expected baud rate the sensor is attempting to decode for NMEA UART input \$GPRMC messages.

1.2.12 `nmea_leap_seconds`

Description:

- Type: Integer ≥ 0
- Default: 0
- Set an integer number of leap seconds that will be added to the UDP timestamp when calculating seconds since 00:00:00 Thursday, 1 January 1970. For Unix Epoch time, this should be set to `0`.

1.2.13 azimuth_window

Description:

- Type: List
- Default: [0,360000]
- Set the visible region of interest of the sensor in millidegrees. Only data from within the specified azimuth window bounds is sent. The value should be provisioned as: [min_bound_millideg, max_bound_millideg]

1.2.14 signal_multiplier

Description:

- Type: Number [0.25, 0.5, 1 ... 3]
- Default: 1
- The value that the signal_multiplier is configured. By default the sensor has a signal multiplier value of 1.

For 2x and 3x multipliers, the `azimuth_window` parameter sets the azimuth window that the lasers will be enabled in.

The higher the signal multiplier value, the smaller the maximum azimuth window can be.

Signal Multiplier Value Max Azimuth Window for 0.25, 0.5 and 1: (Default) 360°, 2: 180°, 3: 120°.

All sensors have equivalent power draw and thermal output when operating at the max azimuth window for a particular signal multiplier value. Therefore, using an azimuth window that is smaller than the maximum allowable azimuth window with a particular signal multiplier value (excluding 1x) can reduce the power draw and thermal output of the sensor.

However, while this can increase the max operating temp of the sensor, it can also degrade the performance at low temps. This discrepancy will be resolved in a future firmware. The table below outlines some example use cases.

1.2.15 udp_profile_lidar

Description:

- Type: Keyword
- Default: "RNG19_RFL8_SIG16_NIR16"
- **Enum:**
 - "RNG19_RFL8_SIG16_NIR16"
 - "RNG19_RFL8_SIG16_NIR16_DUAL"
 - "RNG15_RFL8_NIR8"

- "FUSA_RNG15_RFL8_NIR8_DUAL"
- The configuration of the LIDAR data packets. Valid values are `RNG19_RFL8_SIG16_NIR16` [Default], `RNG19_RFL8_SIG16_NIR16_DUAL`, `RNG15_RFL8_NIR8` and `FUSA_RNG15_RFL8_NIR8_DUAL`.

1.2.16 `udp_profile_imu`

Description:

- Type: Keyword
- Default: "LEGACY"
- Value: "LEGACY"
- The configuration of the IMU data packets. Valid value is `LEGACY`.

1.2.17 `phase_lock_enable`

Description: Whether phase locking is enabled. Refer to Phase Lock Section in the Firmware User Manual for more details on using phase lock.

- Type: Boolean
- Default: False
- Whether phase locking is enabled. Refer to Phase Lock Section in the Firmware User Manual for more details on using phase lock.

1.2.18 `phase_lock_offset`

Description:

- Type: Integer [0 ... 360000]
- Default: 0
- The angle in the Lidar Coordinate Frame that sensors are locked to in millidegrees if phase locking is enabled. Angle is traversed at the top of the second.

1.2.19 `lidar_mode`

Description:

- Type: Keyword
- Default: "1024x10"
- **Enum:**
 - "512x10"

- "1024x10"
 - "2048x10"
 - "512x20"
 - "1024x20"
- The horizontal resolution and rotation rate of the sensor. The effective range of the sensor is increased by 15-20% for every halving of the number of points gathered e.g. 512x10 has 15-20% longer range than 512x20.

1.2.20 `timestamp_mode`

Description:

The method used to timestamp measurements. Valid modes are `TIME_FROM_INTERNAL_OSC`, `TIME_FROM_SYNC_PULSE_IN`, or `TIME_FROM_PTP_1588`.

1.2.21 `multipurpose_io_mode`

Description:

- Type: Keyword
- Default: "OFF"
- **Enum:**
 - "OFF"
 - "INPUT_NMEA_UART"
 - "OUTPUT_FROM_INTERNAL_OSC"
 - "OUTPUT_FROM_SYNC_PULSE_IN"
 - "OUTPUT_FROM_PTP_1588"
 - "OUTPUT_FROM_ENCODER_ANGLE"
- Configure the mode of the MULTIPURPOSE_IO pin. Refer to Time Synchronization section in Firmware user manual for a detailed description of each option.

1.2.22 `operating_mode`

Description:

- Type: Any
- Default: "NORMAL"
- Set `NORMAL` to put the sensor into a normal operating mode or `STANDBY` to put the sensor into a low power (5W) operating mode where the motor does not spin and lasers do not fire.

Note: `auto_start_flag` is deprecated parameter in Firmware 2.4 and later. `auto_start_flag 0` is equivalent to `operating_mode STANDBY` and `auto_start_flag 1` is equivalent to `operating_mode NORMAL`.

1.2.23 `min_range_threshold_cm`

Description:

- Type: Number
- Default:
 - 50 (default)
- Enum:
 - 30
 - 0
- Set 0 or 30 (centimeters) to change the sensor's minimum reported range. Points below the configured minimum range will not be reported. This parameter is present in firmware v3.1 and newer.

Note: When `min_range_threshold_cm` is set to 30 or less, Ouster recommends setting `return_order` to `FARTHEST_TO_NEAREST`.

1.2.24 `return_order`

- Type: Keyword
- Default:
 - `STRONGEST_TO_WEAKEST`
- Enum:
 - `NEAREST_TO_FARTHEST`
 - `FARTHEST_TO_NEAREST`

- This parameter configures how the lidar returns are ordered. The return order can be set by the user (`STRONGEST_TO_WEAKEST` `(default), ``NEAREST_TO_FARTHEST` AND `FARTHEST_TO_NEAREST`). This parameter applies regardless of the configured `udp_profile_lidar` and can be used to configure the only reported return in single return UDP profiles. This parameter is present in firmware v3.1 and newer.

Note: When `min_range_threshold_cm` is set to 30 or less, Ouster recommends setting `return_order` to `FARTHEST_TO_NEAREST`.

1.2.25 `imu_data_format`

- Type: Keyword
- Default:
 - `NORMAL`
- Enum:
 - `NORMAL`
 - `EXTENDED`
- Two valid values for "gyro_fsr" and "accel_fsr" are `NORMAL (Default)`, `EXTENDED`. Please refer to GET /api/v1/sensor/metadata/imu_data_format for more information.

2 HTTP API Reference Guide

This reference guide documents the interface for HTTP API and is accessible via `/api/v1` on the sensor hosted HTTP server.

The sensor can be queried and configured using HTTP requests. This can be done using several different tools such as HTTPie, cURL, Advanced REST Client, etc.

Here is an example using **curl** command:

```
$ curl --request GET --url http://192.0.2.123/api/v1/sensor/metadata/lidar_intrinsics
{
  "lidar_to_sensor_transform": [-1, 0, 0, 0, 0, -1, 0, 0, 0, 0, 1, 38.195, 0, 0, 0, 1]
}
```

Note: All sensor configuration parameters are **CASE SENSITIVE**, please refer to this user manual to make sure the implementation is correct. Additionally all the examples provided for endpoints are tested on Linux, if you are using Mac OS or Windows please adhere to correct command structure.

2.1 Sensor Metadata

2.1.1 GET /api/v1/sensor/metadata/sensor_info

To **GET** the sensor information.

```
GET /api/v1/sensor/metadata/sensor_info HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 285
Content-Type: application/json

{
  "build_date": "2023-1-15T15:56:07Z",
  "build_rev": "v3.0.0",
  "image_rev": "ousteros-image-prod-bootes-v3.0.0+0123456789",
  "initialization_id": 390072,
  "prod_line": "OS-1-128",
  "prod_pn": "860-105010-07",
  "prod_sn": "992244000006",
  "status": "RUNNING"
}
```

Description: Returns JSON-formatted response that includes sensor serial number, product number, FW image revision, and sensor status along with other parameters as shown.

2.1.2 GET /api/v1/sensor/metadata/lidar_data_format

To **GET** the sensor lidar data format.

```
GET /api/v1/sensor/metadata/lidar_data_format HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 724
Content-Type: application/json
Date: Thu, 28 Apr 2022 19:00:38 GMT
Server: nginx

{
  "column_window": [0, 1023],
  "columns_per_frame": 1024,
  "columns_per_packet": 16,
  "pixel_shift_by_row": [
    12, 4, -4, -12, 12, 4, -4,
    -12, 12, 4, -4, -12, 12, 4,
    -4, -12, 12, 4, -4, -12, 12,
    4, -4, -12, 12, 4, -4, -12,
    12, 4, -4, -12, 12, 4, -4,
    -12, 12, 4, -4, -12, 12, 4,
    -4, -12, 12, 4, -4, -12, 12,
    4, -4, -12, 12, 4, -4, -12,
    12, 4, -4, -12, 12, 4, -4,
    -12, 12, 4, -4, -12, 12, 4,
    4, -12, 12, 4, -4, -12, 12,
    4, -4, -12, 12, 4, -4, -12,
    12, 4, -4, -12, 12, 4, -4,
    -12, 12, 4, -4, -12, 12, 4,
    -4, -12],
  "pixels_per_column": 128,
  "udp_profile_imu": "LEGACY",
  "udp_profile_lidar": "RNG19_RFL8_SIG16_NIR16_DUAL"
}
```

Description: Returns JSON-formatted response that describes the structure of a lidar packet.

- **columns_per_frame:** Number of measurement columns per frame. This can be 512, 1024 or 2048 depending upon the set lidar mode.
- **columns_per_packet:** Number of measurement blocks contained in a single lidar packet. **Note:** This is not user configurable.
- **pixel_shift_by_row:** Offset in terms of pixel count. Can be used to destagger image. Varies by lidar mode. Length of this array is equal to the number of channels of the sensor.
- **pixels_per_column:** Number of channels of the sensor.

- `column_window`: Index of measurement blocks that are active. Default is [0, lidar_mode-1], e.g. [0,1023]. If there is an azimuth window set, this parameter will reflect which measurement blocks of data are within the region of interest.
- `udp_profile_lidar`: Lidar data profile format. Defaults to single return profile (RNG19_RFL8_SIG16_NIR16).
- `udp_profile_imu`: IMU data profile format. Default is `LEGACY`.

Note: This command only works when the sensor is in **RUNNING** status.

2.1.3 GET /api/v1/sensor/metadata/imu_data_format

To `GET` the sensor IMU data format. Valid values are `NORMAL` (Default) and `EXTENDED`.

```
GET /api/v1/sensor/metadata/imu_data_format HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 45
Content-Type: application/json
Date: Mon, 04 Mar 2024 20:07:50 GMT
Server: nginx

{
  "accel_fsr": "NORMAL",
  "gyro_fsr": "NORMAL"
}
```

Description:

- `"accel_fsr"` parameter refers to the Full Scale Range of the accelerometer.
- `"gyro_fsr"` parameter relates to the Full Scale Range of the gyroscope.

Note: User can run `DELETE /api/v1/sensor/config` command to configure `imu_data_format` back to default i.e., `NORMAL`. For more information on the IMU specifications refer to Sensor Datasheet.

2.1.4 GET /api/v1/sensor/metadata/beam_intrinsics

To `GET` the sensor beam intrinsics.

```
GET /api/v1/sensor/metadata/beam_intrinsics HTTP/1.1
Host: 192.0.2.123
```



```

HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 1895
Content-Type: application/json

{
  "beam_altitude_angles": [
    20.38, 20.12, 19.79, 19.45, 19.14, 18.85, 18.55, 18.2, 17.86, 17.58, 17.27, 16.93,
    16.58, 16.29, 15.98, 15.61, 15.27, 14.97, 14.66, 14.3, 13.96, 13.65, 13.33, 12.97,
    12.62, 12.31, 11.98, 11.63, 11.27, 10.96, 10.63, 10.26, 9.91, 9.59, 9.26, 8.89,
    8.54, 8.21, 7.87, 7.52, 7.15, 6.82, 6.47, 6.11, 5.76, 5.42, 5.08, 4.73, 4.36, 4.03,
    3.66, 3.31, 2.96, 2.62, 2.27, 1.91, 1.55, 1.22, 0.85, 0.51, 0.16, -0.2, -0.55, -0.91,
    -1.26, -1.62, -1.96, -2.3, -2.66, -3.02, -3.36, -3.72, -4.07, -4.42, -4.77, -5.11,
    -5.46, -5.82, -6.16, -6.49, -6.85, -7.21, -7.55, -7.88, -8.23, -8.59, -8.93, -9.25,
    -9.6, -9.96, -10.31, -10.63, -10.96, -11.32, -11.67, -11.97, -12.31, -12.68, -13,
    -13.32, -13.64, -14, -14.33, -14.63, -14.96, -15.31, -15.64, -15.94, -16.26,
    -16.62, -16.93, -17.22, -17.54, -17.9, -18.22, -18.49, -18.8, -19.16, -19.47,
    -19.73, -20.04, -20.39, -20.7, -20.94, -21.25, -21.6, -21.9, -22.14
  ],
  "beam_azimuth_angles": [
    4.24, 1.41, -1.42, -4.23, 4.23, 1.41, -1.41, -4.23, 4.23, 1.41, -1.41, -4.21, 4.23,
    1.42, -1.4, -4.23, 4.24, 1.41, -1.4, -4.23, 4.24, 1.42, -1.4, -4.22, 4.23, 1.41,
    -1.41, -4.22, 4.23, 1.42, -1.4, -4.22, 4.24, 1.41, -1.4, -4.23, 4.23, 1.41, -1.41,
    -4.22, 4.23, 1.41, -1.41, -4.23, 4.23, 1.4, -1.42, -4.23, 4.23, 1.41, -1.42, -4.23,
    4.23, 1.4, -1.42, -4.24, 4.22, 1.41, -1.43, -4.24, 4.22, 1.4, -1.42, -4.24, 4.22,
    1.4, -1.42, -4.23, 4.22, 1.4, -1.4, -4.24, 4.22, 1.4, -1.42, -4.24, 4.22, 1.41,
    -1.41, -4.22, 4.22, 1.39, -1.42, -4.23, 4.22, 1.41, -1.41, -4.22, 4.23, 1.41,
    -1.41, -4.23, 4.23, 1.41, -1.41, -4.22, 4.23, 1.41, -1.41, -4.22, 4.22, 1.41,
    -1.41, -4.22, 4.23, 1.41, -1.4, -4.23, 4.22, 1.41, -1.41, -4.23, 4.22, 1.4, -1.41,
    -4.23, 4.22, 1.4, -1.41, -4.24, 4.22, 1.4, -1.42, -4.24, 4.22, 1.4, -1.42, -4.23
  ],
  "beam_to_lidar_transform": [ 1, 0, 0, 15.805999755859375, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1 ],
  "lidar_origin_to_beam_origin_mm": 15.8059998
}

```

Description: Returns JSON-formatted beam altitude and azimuth offsets, in degrees. Length of arrays is equal to the number of channels in the sensor. Also returns distance between lidar origin and beam origin in mm, to be used for point cloud calculations.

2.1.5 GET /api/v1/sensor/metadata/imu_intrinsics

To **GET** the sensor imu intrinsics

```

GET /api/v1/sensor/metadata/imu_intrinsics HTTP/1.1
Host: 192.0.2.123

```

```

HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 91
Content-Type: application/json

{
  "imu_to_sensor_transform": [1, 0, 0, 6.253, 0, 1, 0, -11.775, 0, 0, 1, 7.645, 0, 0, 0, 1]
}

```

(continues on next page)

(continued from previous page)

}

Description: Returns JSON-formatted IMU transformation matrix needed to transform to the Sensor Coordinate Frame.

2.1.6 GET /api/v1/sensor/metadata/lidar_intrinsics

To **GET** the sensor lidar intrinsics

```
GET /api/v1/sensor/metadata/lidar_intrinsics HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 86
Content-Type: application/json

{
  "lidar_to_sensor_transform": [-1, 0, 0, 0, 0, -1, 0, 0, 0, 0, 1, 38.195, 0, 0, 0, 1]
}
```

Description: Returns JSON-formatted lidar transformation matrix needed to transform to the Sensor Coordinate Frame.

2.1.7 GET /api/v1/sensor/metadata/calibration_status

To **GET** the sensor calibration status.

```
GET /api/v1/sensor/metadata/calibration_status HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 69
Content-Type: application/json

{
  "reflectivity":
  {
    "timestamp": "2022-11-18T20:31:06",
    "valid": true
  }
}
```

Description: Returns JSON formatted calibration status of the sensor reflectivity. **valid**: true/false depending on calibration status. **timestamp**: if valid is true; time at which the calibration was completed.

2.1.8 GET /api/v1/sensor/config

To **GET** all sensor configuration parameters.

```
GET /api/v1/sensor/config HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 829
Content-Type: application/json
Date: Mon, 04 Mar 2024 20:14:33 GMT
Server: nginx

{
  "accel_fsr": "NORMAL",
  "azimuth_window": [
    0,
    360000
  ],
  "columns_per_packet": 16,
  "gyro_fsr": "NORMAL",
  "lidar_mode": "1024x10",
  "min_range_threshold_cm": 0,
  "multipurpose_io_mode": "OFF",
  "nmea_baud_rate": "BAUD_9600",
  "nmea_ignore_valid_char": 0,
  "nmea_in_polarity": "ACTIVE_HIGH",
  "nmea_leap_seconds": 0,
  "operating_mode": "NORMAL",
  "phase_lock_enable": false,
  "phase_lock_offset": 0,
  "return_order": "STRONGEST_TO_WEAKEST",
  "signal_multiplier": 0.25,
  "sync_pulse_in_polarity": "ACTIVE_HIGH",
  "sync_pulse_out_angle": 360,
  "sync_pulse_out_frequency": 1,
  "sync_pulse_out_polarity": "ACTIVE_HIGH",
  "sync_pulse_out_pulse_width": 10,
  "timestamp_mode": "TIME_FROM_INTERNAL_OSC",
  "udp_dest": "192.0.2.123",
  "udp_port_imu": 7503,
  "udp_port_lidar": 7502,
  "udp_profile_imu": "LEGACY",
  "udp_profile_lidar": "RNG19_RFL8_SIG16_NIR16_DUAL"
}
```

Description: Please refer to [Description](#) section for a detailed description of sensor configurable parameters.

2.1.9 POST /api/v1/sensor/config - Multiple configuration parameters

Multiple configuration parameters can be set at one time using this command. All of the specified sensor configuration parameters must be set successfully for the **POST** request to succeed, otherwise none of the specified sensor configuration parameters will be configured, and an error will be returned.

Note: After POST `api/v1/sensor/config` request is received successfully, the sensor will reinitialize automatically to make the new configuration active, and the config settings are persisted across power cycles.

Example 1

```
POST /api/v1/sensor/config HTTP/1.1
Host: 192.0.2.123

{
  "accel_fsr": "NORMAL",
  "azimuth_window": [
    0,
    360000
  ],
  "columns_per_packet": 16,
  "gyro_fsr": "NORMAL",
  "lidar_mode": "1024x10",
  "min_range_threshold_cm": 0,
  "multipurpose_io_mode": "OFF",
  "nmea_baud_rate": "BAUD_9600",
  "nmea_ignore_valid_char": 0,
  "nmea_in_polarity": "ACTIVE_HIGH",
  "nmea_leap_seconds": 0,
  "operating_mode": "NORMAL",
  "phase_lock_enable": false,
  "phase_lock_offset": 0,
  "return_order": "STRONGEST_TO_WEAKEST",
  "signal_multiplier": 0.25,
  "sync_pulse_in_polarity": "ACTIVE_HIGH",
  "sync_pulse_out_angle": 360,
  "sync_pulse_out_frequency": 1,
  "sync_pulse_out_polarity": "ACTIVE_HIGH",
  "sync_pulse_out_pulse_width": 10,
  "timestamp_mode": "TIME_FROM_INTERNAL_OSC",
  "udp_dest": "10.34.24.163",
  "udp_port_imu": 7503,
  "udp_port_lidar": 7502,
  "udp_profile_imu": "LEGACY",
  "udp_profile_lidar": "RNG15_RFL8_NIR8"
}
```

```
HTTP/1.1 204 No Content
Connection: keep-alive
Date: Thu, 01 Jan 1970 00:37:41 GMT
Server: nginx
```

To verify, the user can run `GET /api/v1/sensor/config` after `POST` command:

```
{
  "accel_fsr": "NORMAL",
  "azimuth_window": [
    0,
    360000
  ],
  "columns_per_packet": 16,
  "gyro_fsr": "NORMAL",
  "lidar_mode": "1024x10",
  "min_range_threshold_cm": 0,
  "multipurpose_io_mode": "OFF",
  "nmea_baud_rate": "BAUD_9600",
  "nmea_ignore_valid_char": 0,
  "nmea_in_polarity": "ACTIVE_HIGH",
  "nmea_leap_seconds": 0,
  "operating_mode": "NORMAL",
  "phase_lock_enable": false,
  "phase_lock_offset": 0,
  "return_order": "STRONGEST_TO_WEAKEST",
  "signal_multiplier": 0.25,
  "sync_pulse_in_polarity": "ACTIVE_HIGH",
  "sync_pulse_out_angle": 360,
  "sync_pulse_out_frequency": 1,
  "sync_pulse_out_polarity": "ACTIVE_HIGH",
  "sync_pulse_out_pulse_width": 10,
  "timestamp_mode": "TIME_FROM_INTERNAL_OSC",
  "udp_dest": "10.34.24.163",
  "udp_port_imu": 7503,
  "udp_port_lidar": 7502,
  "udp_profile_imu": "LEGACY",
  "udp_profile_lidar": "RNG15_RFL8_NIR8"
}
```

2.1.10 GET /api/v1/sensor/config/operating_mode

Any configuration parameter can be queried by using a URI with the format `/api/v1/sensor/config/parameter_name`.

Example 1

```
GET /api/v1/sensor/config/operating_mode HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 204 No Content
Connection: keep-alive
Date: Thu, 01 Jan 1970 00:37:41 GMT
Server: nginx

"NORMAL"
```

2.1.11 POST /api/v1/sensor/config/operating_mode

Any configuration parameter can be set by using a URI with the format `/api/v1/sensor/config/parameter_name`. In this example the operating mode is changed.

Note: After a POST `api/v1/sensor/config` request is received successfully, the sensor will reinitialize automatically to make the new configuration active, and the config settings are persisted across power cycles.

Example 1

```
POST /api/v1/sensor/config/operating_mode HTTP/1.1
Host: 192.0.2.123
```

```
"STANDBY"
```

```
HTTP/1.1 204 No Content
Connection: keep-alive
Date: Thu, 01 Jan 1970 00:37:41 GMT
Server: nginx
```

2.1.12 DELETE /api/v1/sensor/config

Note: This API command resets all sensor configuration to the default state, including static IP address settings, and restarts the sensor.

```
DELETE /api/v1/sensor/config HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 204 No Content
Connection: keep-alive
Date: Mon, 10 Jul 2023 17:12:47 GMT
Server: nginx
```

2.1.13 GET /api/v1/sensor/metadata

To **GET** the sensor metadata information.

```
GET /api/v1/sensor/metadata HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 4009
Content-Type: application/json
```

```
{
  "beam_intrinsics": {
    "beam_altitude_angles": [
      20.38, 20.12, 19.79, 19.45, 19.14, 18.85, 18.55, 18.2, 17.86, 17.58, 17.27, 16.93,
      16.58, 16.29, 15.98, 15.61, 15.27, 14.97, 14.66, 14.3, 13.96, 13.65, 13.33, 12.97,
      12.62, 12.31, 11.98, 11.63, 11.27, 10.96, 10.63, 10.26, 9.91, 9.59, 9.26, 8.89,
      8.54, 8.21, 7.87, 7.52, 7.15, 6.82, 6.47, 6.11, 5.76, 5.42, 5.08, 4.73, 4.36, 4.03,
      3.66, 3.31, 2.96, 2.62, 2.27, 1.91, 1.55, 1.22, 0.85, 0.51, 0.16, -0.2, -0.55, -0.91,
      -1.26, -1.62, -1.96, -2.3, -2.66, -3.02, -3.36, -3.72, -4.07, -4.42, -4.77, -5.11,
      -5.46, -5.82, -6.16, -6.49, -6.85, -7.21, -7.55, -7.88, -8.23, -8.59, -8.93, -9.25,
      -9.6, -9.96, -10.31, -10.63, -10.96, -11.32, -11.67, -11.97, -12.31, -12.68, -13,
      -13.32, -13.64, -14, -14.33, -14.63, -14.96, -15.31, -15.64, -15.94, -16.26,
      -16.62, -16.93, -17.22, -17.54, -17.9, -18.22, -18.49, -18.8, -19.16, -19.47,
      -19.73, -20.04, -20.39, -20.7, -20.94, -21.25, -21.6, -21.9, -22.14
    ],
    "beam_azimuth_angles": [
      4.24, 1.41, -1.42, -4.23, 4.23, 1.41, -1.41, -4.23, 4.23, 1.41, -1.41, -4.21, 4.23,
      1.42, -1.4, -4.23, 4.24, 1.41, -1.4, -4.23, 4.24, 1.42, -1.4, -4.22, 4.23, 1.41,
      -1.41, -4.22, 4.23, 1.42, -1.4, -4.22, 4.24, 1.41, -1.4, -4.23, 4.23, 1.41, -1.41,
      -4.22, 4.23, 1.41, -1.41, -4.23, 4.23, 1.4, -1.42, -4.23, 4.23, 1.41, -1.42, -4.23,
      4.23, 1.4, -1.42, -4.24, 4.22, 1.41, -1.43, -4.24, 4.22, 1.4, -1.42, -4.24, 4.22,
      1.4, -1.42, -4.23, 4.22, 1.4, -1.4, -4.24, 4.22, 1.4, -1.42, -4.24, 4.22, 1.41,
      -1.41, -4.22, 4.22, 1.39, -1.42, -4.23, 4.22, 1.41, -1.41, -4.22, 4.23, 1.41,
      -1.41, -4.23, 4.23, 1.41, -1.41, -4.22, 4.23, 1.41, -1.41, -4.22, 4.22, 1.41,
      -1.41, -4.22, 4.23, 1.41, -1.4, -4.23, 4.22, 1.41, -1.41, -4.23, 4.22, 1.4, -1.41,
      -4.23, 4.22, 1.4, -1.41, -4.24, 4.22, 1.4, -1.42, -4.24, 4.22, 1.4, -1.42, -4.23
    ]
  }
}
```

(continues on next page)


```

    ],
    "beam_to_lidar_transform": [ 1, 0, 0, 15.805999755859375, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1 ],
    "lidar_origin_to_beam_origin_mm": 15.8059998
  }
  "calibration_status": {
    "reflectivity": {
      "timestamp": "2022-11-18T20:31:06",
      "valid": true
    }
  },
  "config_params": {
    "azimuth_window": [
      0,
      360000
    ],
    "columns_per_packet": 16,
    "lidar_mode": "1024x10",
    "multipurpose_io_mode": "OFF",
    "nmea_baud_rate": "BAUD_9600",
    "nmea_ignore_valid_char": 0,
    "nmea_in_polarity": "ACTIVE_HIGH",
    "nmea_leap_seconds": 0,
    "operating_mode": "NORMAL",
    "phase_lock_enable": false,
    "phase_lock_offset": 0,
    "signal_multiplier": 1,
    "sync_pulse_in_polarity": "ACTIVE_HIGH",
    "sync_pulse_out_angle": 360,
    "sync_pulse_out_frequency": 1,
    "sync_pulse_out_polarity": "ACTIVE_HIGH",
    "sync_pulse_out_pulse_width": 10,
    "timestamp_mode": "TIME_FROM_INTERNAL_OSC",
    "udp_dest": "169.254.225.4",
    "udp_port_imu": 7503,
    "udp_port_lidar": 7502,
    "udp_profile_imu": "LEGACY",
    "udp_profile_lidar": "RNG19_RFL8_SIG16_NIR16_DUAL"
  },
  "imu_intrinsic": {
    "imu_to_sensor_transform": [
      1, 0, 0, 6.253, 0, 1, 0, -11.775, 0, 0, 1, 7.645, 0, 0, 0, 1
    ]
  },
  "lidar_data_format": {
    "column_window": [
      0,
      1023
    ],
    "columns_per_frame": 1024,
    "columns_per_packet": 16,
    "pixel_shift_by_row": [12, 4, -4, -12, 12,
      4, -4, -12, 12, 4, -4, -12, 12, 4, -4,
      -12, 12, 4, -4, -12, 12, 4, -4, -12, 12,
      4, -4, -12, 12, 4, -4, -12, 12, 4, -4,
    ]
  }
}

```

```

-12, 12, 4, -4, -12, 12, 4, -4, -12, 12,
4, -4, -12, 12, 4, -4, -12, 12, 4, -4,
-12, 12, 4, -4, -12, 12, 4, -4, -12, 12,
4, -4, -12, 12, 4, -4, -12, 12, 4, -4,
-12, 12, 4, -4, -12, 12, 4, -4, -12, 12,
4, -4, -12, 12, 4, -4, -12, 12, 4, -4,
-12, 12, 4, -4, -12, 12, 4, -4, -12, 12,
4, -4, -12, 12, 4, -4, -12, 12, 4, -4,
-12, 12, 4, -4, -12, 12, 4, -4, -12, 12,
4, -4, -12
],
"pixels_per_column": 128,
"udp_profile_imu": "LEGACY",
"udp_profile_lidar": "RNG19_RFL8_SIG16_NIR16_DUAL"
},

"lidar_intrinsics": {
  "lidar_to_sensor_transform": [
    -1, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 1, 38.195, 0, 0, 0, 1
  ]
},
"sensor_info": {
  "build_date": "2023-1-15T15:56:07Z",
  "build_rev": "v3.0.0",
  "image_rev": "ousteros-image-prod-bootes-v3.0.0+0123456789",
  "initialization_id": 390079,
  "prod_line": "OS-1-128",
  "prod_pn": "860-105010-07",
  "prod_sn": "992244000006",
  "status": "RUNNING"
}
}

```

2.2 User Editable Data

A user configurable data field is made available in firmware v2.5 and later.

This field can be used for a number of purposes such as storing specific information about the sensor, qualifying a sensor, calibration data, or any other information. Please refer to User Data Field section in the Firmware User Manual for more information.

Additional Information:

- **Valid values for UED:** Empty string or string containing non-binary ASCII and/or Unicode characters.
- **Size limit for UED string:** 128KB with 1KB = 1024bytes (Total = 131,072 bytes)

2.2.1 GET /api/v1/user/data

Retrieve the current value of 'user-data-field', "" is returned by default.

```
GET /api/v1/user/data HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 15
Content-Type: application/json
Date: Thu, 01 Jan 1970 01:40:23 GMT
Server: nginx

""
```

2.2.2 PUT /api/v1/user/data

Puts a "content" in the `user data field`. In the example shown below we will use **"Ouster sensor"** as the content to be put in the `user data field`.

Default `data policy` for `PUT` request on user editable data is `clear_on_config_delete`. If you would like to persist the value in the data field please see [PUT /api/v1/user/data?policy=keep_on_config_delete](#).

```
PUT /api/v1/user/data HTTP/1.1
Host: 192.0.2.123
```

```
"Ouster sensor"
```

```
HTTP/1.1 204 No Content
Connection: keep-alive
Date: Thu, 01 Jan 1970 01:39:16 GMT
Server: nginx
```

To verify: Run `GET /api/v1/user/data`

```
GET /api/v1/user/data HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 15
Content-Type: application/json
Date: Thu, 01 Jan 1970 01:05:14 GMT
Server: nginx

"Ouster sensor"
```

2.2.3 DELETE /api/v1/user/data

Deletes the current value ("content") in the 'user-data-field'.

```
DELETE /api/v1/user/data HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 204 No Content
Connection: keep-alive
Date: Thu, 01 Jan 1970 01:05:32 GMT
Server: nginx
```

To verify: Run *GET /api/v1/user/data*

```
GET /api/v1/user/data HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 2
Content-Type: application/json
Date: Thu, 01 Jan 1970 01:05:14 GMT
Server: nginx
```

```
""
```

2.2.4 Optional Parameters - data policy

The policy key maps to the active policy as applied with `PUT api/v1/user/data?policy=<policy_str>`.

`<policy_str>` have the following options available:

- `clear_on_config_delete` by default
- `keep_on_config_delete`

Note: `Data policy` has no effect on the content of the User Editable Data field.

PUT /api/v1/user/data?policy=clear_on_config_delete

```
PUT /api/v1/user/data?policy=clear_on_config_delete HTTP/1.1
Host: 192.0.2.123
```

```
"Ouster sensor"
```

```
HTTP/1.1 204 No Content
Connection: keep-alive
Date: Thu, 01 Jan 1970 00:43:08 GMT
Server: nginx
```

PUT /api/v1/user/data?policy=keep_on_config_delete

When `keep_on_config_delete` has been applied, the data in the user editable data field is persisted regardless of any sensor configuration resets or shutdown. If the user needs to reset this field then please run `DELETE /api/v1/user/data`.

```
PUT /api/v1/user/data?policy=keep_on_config_delete HTTP/1.1
Host: 192.0.2.123

"Ouster Sensor"
```

```
HTTP/1.1 204 No Content
Connection: keep-alive
Date: Thu, 01 Jan 1970 00:43:08 GMT
Server: nginx
```

2.2.5 Optional Parameters - include_metadata

Same as nominal `GET` but returns a JSON dictionary of the form `{ "value": str, "policy": str }` where the value key maps to the nominal value returned by `GET` with no arguments.

Note: `include_metadata` has no effect on the User Editable Data field.

This feature lets user to query the user editable data field to get `policy` and `value` when `include_metadata` is set to `true/1` and only the `value` when `include_matadata` is set to `false/0`

GET /api/v1/user/data?include_metadata=true

Returns a JSON dictionary of the form `{ "value": str, "policy": str }`.

```
GET /api/v1/user/data?include_metadata=true HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 61
Content-Type: application/json
Date: Thu, 01 Jan 1970 00:20:48 GMT
Server: nginx

{
  "policy": "keep_on_config_delete",
  "value": "ouster sensor"
}
```

GET /api/v1/user/data?include_metadata=false

Returns only the value of the user data.

```
GET /api/v1/user/data?include_metadata=false HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 15
Content-Type: application/json
Date: Thu, 01 Jan 1970 00:21:41 GMT
Server: nginx

"Ouster sensor"
```

2.3 System

2.3.1 POST /api/v1/system/restart

Restarts the sensor. This command is present in firmware version v3.1 and newer.

Warning: Please contact Ouster support if you find the need to use this command.

```
POST /api/v1/system/restart HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 204 No Content
Connection: keep-alive
Date: Thu, 14 Mar 2024 20:33:01 GMT
Server: nginx
```

2.3.2 GET /api/v1/system/firmware

To **GET** the firmware version of the sensor.

```
GET /api/v1/system/firmware HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 82
Content-Type: application/json
Date: Thu, 01 Jan 1970 00:27:54 GMT
Server: nginx

{
  "commit_pending": false,
  "fw": "ousteros-image-prod-aries-v2.5.2+20230714195410"
}
```

>**json string fw** Running firmware image name and version.

2.3.3 POST /api/v1/system/firmware

To update the sensor firmware.

Example:

```
curl -vH 'content-type: application/octet-stream' --data-binary @../Downloads/ousteros-image-prod-aries-v2.5.x+20230607131746.staging.img http://192.0.2.123/api/v1/system/firmware
```

Response:

```
* Trying 192.0.2.123:80...
* TCP_NODELAY set
* Connected to 192.0.2.123 (192.0.2.123) port 80 (#0)
> POST /api/v1/system/firmware HTTP/1.1
> Host: 192.0.2.123
> User-Agent: curl/7.68.0
> Accept: */*
> content-type: application/octet-stream
> Content-Length: 42755180
> Expect: 100-continue
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 100 Continue
* We are completely uploaded and fine
* Mark bundle as not supporting multiuse
< HTTP/1.1 204 No Content
< Server: nginx
< Date: Wed, 14 Jun 2023 23:38:08 GMT
< Connection: keep-alive
<
* Connection #0 to host 192.0.2.123 left intact
```

2.3.4 GET /api/v1/system/network

To **GET** the system network configuration.

```
GET /api/v1/system/network HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 256
Content-Type: application/json

{
  "carrier": true,
  "duplex": "full",
  "ethaddr": "bc:0f:a7:00:84:17",
  "hostname": "192.0.2.123",
  "ipv4": {
    "link_local": "192.0.2.123/16",
    "override": null
  },
  "ipv6": {
    "link_local": "fe80::be0f:a7ff:fe00:8417/64"
  },
  "speed": 1000,
  "speed_override": null
}
```

- **>json boolean carrier:** State of Ethernet link, **true** when physical layer is connected.
- **>json string duplex:** Duplex mode of Ethernet link, **half** or **full**.
- **>json string ethaddr:** Ethernet hardware (MAC) address.
- **>json string hostname:** Hostname of the sensor, also used when requesting *DHCP* address and registering mDNS hostname.
- **>json object ipv4:** See *ipv4 object*
- **>json string ipv6.link_local:** Link-local IPv6 address.
- **>json integer speed:** Ethernet physical layer speed in Mbps, should be 1000 Mbps.

2.3.5 GET /api/v1/system/network/ipv4

To **GET** the IPv4 network configuration.

```
GET /api/v1/system/network/ipv4 HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 53
```

(continues on next page)

(continued from previous page)

```
Content-Type: application/json
```

```
{  
  "link_local": "192.0.2.123/16",  
  "override": null  
}
```

- **>json string addr:** Current global or private IPv4 address.
- **>json string link_local:** Link-local IPv4 address.
- **>json string override:** Static IP override value, this should match `addr`. This value will be `null` when unset and operating in *DHCP* or *link-local* modes.

2.3.6 GET /api/v1/system/network/ipv4/override

To **GET** the current IPv4 static IP address override.

```
GET /api/v1/system/network/ipv4/override HTTP/1.1  
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK  
Connection: keep-alive  
Content-Length: 4  
Content-Type: application/json  
  
null
```

- **>json string** Static IP override value, this should match `addr`. This value will be `null` when unset and operating in *DHCP* mode.

2.3.7 PUT /api/v1/system/network/ipv4/override

To override the default dynamic behavior and set a static IP address. Only a valid Unicast IPv4 address can be specified when using **PUT** command.

Note: The sensor will reset the network configuration after a short sub second delay (to allow for the HTTP response to be sent). After this delay the sensor will only be reachable on the newly set IPv4 address.

The sensor needs to be reachable either by *link-local* or dynamic *DHCP* configuration or by an existing static IP override from the host reconfiguring the sensor. The sensor may be reset back to using DHCP by **DELETE** ing the sensor configuration.

Warning: If an unreachable network address is set, the sensor will become unreachable. Tools such as `avahi-browse`, `dns-sd`, or `mDNS browser` can help with finding a sensor on a network. Static

IP override should only be used in special use cases. DHCP configuration is recommended where possible.

```
PUT /api/v1/system/network/ipv4/override HTTP/1.1
Host: 192.0.2.123
```

```
"192.0.2.231"
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 17
Content-Type: application/json
Date: Thu, 30 Mar 2023 04:30:30 GMT
Server: nginx
```

```
"192.0.2.231"
```

- **<json string:** Static IP override value with subnet mask
- **>json string:** Static IP override value that system will set after a short delay.

Note: Sensor can be accessed on the sensor's self-assigned link-local IPv4 or IPv6 addresses, in case the sensor becomes unreachable on the configured network due to a misconfiguration. To discover the self-assigned link-local IPv4 or IPv6 addresses for a sensor one can use a network sniffer (such as Wireshark) on the same network segment as the sensor.

2.3.8 DELETE /api/v1/system/network/ipv4/override

To delete the static IP override value and return to dynamic configuration (*DHCP*).

Note: The sensor will reset the network configuration after a short sub second delay (to allow for the HTTP response to be sent). After this delay the sensor will only be reachable on the newly set IPv4 address.

The sensor may be unreachable for several seconds while a *link-local* lease is obtained from the network or client machine.

```
DELETE /api/v1/system/network/ipv4/override HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 204 No Content
Connection: keep-alive
Date: Thu, 01 Jan 1970 19:12:15 GMT
Server: nginx
```

2.3.9 GET /api/v1/system/network/speed_override

Two options `null` (default) and `100`.

Note: Only valid for sensors with automotive ethernet (T1).

Example

```
GET /api/v1/system/network/speed_override HTTP/1.1
Content-Type: application/json
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 4
Content-Type: application/json
Date: Thu, 28 Apr 2022 17:48:51 GMT
Server: nginx

null
```

2.3.10 PUT /api/v1/system/network/speed_override

Note: Only valid for sensors with automotive ethernet (T1).

Warning: Only run this command if you have the ability to configure your networking hardware between 1000BASE-T1 and 100BASE-T1. If you do not have the configuration option available, you will no longer be able to communicate with the sensor. Please refer to an [Example 100Base-T1 Connector](#).

Two options `1000` (default) and `100`. However, user can only use `PUT` command to set speed-override to `100`. In order to revert back to `1000` (default), please run the `DELETE /api/v1/system/network/speed_override`.

Example

```
PUT /api/v1/system/network/speed_override HTTP/1.1
Content-Type: application/json
Host: 192.0.2.123
```

```
100
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 3
Content-Type: application/json
```

(continues on next page)

(continued from previous page)

```
Date: Tue, 18 Jul 2023 19:34:27 GMT
Server: nginx

100
```

2.3.11 DELETE /api/v1/system/network/speed_override

Note: Only valid for sensors with automotive ethernet (T1).

To reset it back to `default` i.e., `1000`

```
DELETE /api/v1/system/network/speed_override HTTP/1.1
Content-Type: application/json
Host: 192.0.2.123
```

```
HTTP/1.1 204 No Content
Connection: keep-alive
Date: Tue, 18 Jul 2023 19:37:52 GMT
Server: nginx
```

2.4 Time

2.4.1 GET /api/v1/time

To `GET` the system time configuration for all timing components of the sensor.

```
GET /api/v1/time HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 2484
Content-Type: application/json

{
  "ptp": {
    "current_data_set": {
      "mean_path_delay": 0.0,
      "offset_from_master": 0.0,
      "steps_removed": 0
    },
    "parent_data_set": {
      "gm_clock_accuracy": 254,
      "gm_clock_class": 255,
      "gm_offset_scaled_log_variance": 65535,
      "grandmaster_identity": "bc0fa7.ffff.008417",
      "grandmaster_priority1": 128,

```

(continues on next page)

```

    "grandmaster_priority2": 128,
    "observed_parent_clock_phase_change_rate": 2147483647,
    "observed_parent_offset_scaled_log_variance": 65535,
    "parent_port_identity": "bc0fa7.ffff.008417-0",
    "parent_stats": 0
  },
  "port_data_set": {
    "announce_receipt_timeout": 3,
    "delay_mechanism": 1,
    "log_announce_interval": 1,
    "log_min_delay_req_interval": 0,
    "log_min_pdelay_req_interval": 0,
    "log_sync_interval": 0,
    "peer_mean_path_delay": 0,
    "port_identity": "bc0fa7.ffff.008417-1",
    "port_state": "LISTENING",
    "version_number": 2
  },
  "profile": "default",
  "time_properties_data_set": {
    "current_utc_offset": 37,
    "current_utc_offset_valid": 0,
    "frequency_traceable": 0,
    "leap59": 0,
    "leap61": 0,
    "ptp_timescale": 1,
    "time_source": 160,
    "time_traceable": 0
  },
  "time_status_np": {
    "cumulative_scaled_rate_offset": 0.0,
    "gm_identity": "bc0fa7.ffff.008417",
    "gm_present": false,
    "gm_time_base_indicator": 0,
    "ingress_time": 0,
    "last_gm_phase_change": "0x0000'0000000000000000.0000",
    "master_offset": 0,
    "scaled_last_gm_phase_change": 0
  }
},
"sensor": {
  "multipurpose_io": {
    "mode": "OFF",
    "nmea": {
      "baud_rate": "BAUD_9600",
      "diagnostics": {
        "decoding": {
          "date_decoded_count": 0,
          "last_read_message": "",
          "not_valid_count": 0,
          "utc_decoded_count": 0
        },
      },
      "io_checks": {
        "bit_count": 1,

```

```

        "bit_count_unfiltered": 0,
        "char_count": 0,
        "start_char_count": 0
    }
},
"ignore_valid_char": 0,
"leap_seconds": 0,
"locked": 0,
"polarity": "ACTIVE_HIGH"
},
"sync_pulse_out": {
    "angle_deg": 360,
    "frequency_hz": 1,
    "polarity": "ACTIVE_HIGH",
    "pulse_width_ms": 10
}
},
"sync_pulse_in": {
    "diagnostics": {
        "count": 1,
        "count_unfiltered": 0,
        "last_period_nsec": 0
    },
    "locked": 0,
    "polarity": "ACTIVE_HIGH"
},
"timestamp": {
    "mode": "TIME_FROM_INTERNAL_OSC",
    "time": 297.397987312,
    "time_options": {
        "internal_osc": 297,
        "ptp_1588": 1651197874,
        "sync_pulse_in": 1
    }
}
},
"system": {
    "monotonic": 30131.822811617,
    "realtime": 1651197874.3271277,
    "tracking": {
        "frequency": -9.558,
        "last_offset": 0.0,
        "leap_status": "not synchronised",
        "ref_time_utc": 0.0,
        "reference_id": "00000000",
        "remote_host": "",
        "residual_frequency": 0.0,
        "rms_offset": 0.0,
        "root_delay": 1.0,
        "root_dispersion": 1.0,
        "skew": 0.0,
        "stratum": 0,
        "system_time_offset": 1e-09,
        "update_interval": 0.0
    }
}

```

(continued from previous page)

```
}  
  }  
}
```

- **>json string:** See sub objects for details.

2.4.2 GET /api/v1/time/sensor

To **GET** the sensor time information.

```
GET /api/v1/time/sensor HTTP/1.1  
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK  
Connection: keep-alive  
Content-Length: 775  
Content-Type: application/json  
  
{  
  "multipurpose_io": {  
    "mode": "OFF",  
    "nmea": {  
      "baud_rate": "BAUD_9600",  
      "diagnostics": {  
        "decoding": {  
          "date_decoded_count": 0,  
          "last_read_message": "",  
          "not_valid_count": 0,  
          "utc_decoded_count": 0  
        },  
        "io_checks": {  
          "bit_count": 1,  
          "bit_count_unfiltered": 0,  
          "char_count": 0,  
          "start_char_count": 0  
        }  
      },  
      "ignore_valid_char": 0,  
      "leap_seconds": 0,  
      "locked": 0,  
      "polarity": "ACTIVE_HIGH"  
    },  
    "sync_pulse_out": {  
      "angle_deg": 360,  
      "frequency_hz": 1,  
      "polarity": "ACTIVE_HIGH",  
      "pulse_width_ms": 10  
    }  
  },  
  "sync_pulse_in": {  
    "diagnostics": {  
      "count": 1,  

```

(continues on next page)

(continued from previous page)

```
        "count_unfiltered": 0,
        "last_period_nsec": 0
    },
    "locked": 0,
    "polarity": "ACTIVE_HIGH"
},
"timestamp": {
    "mode": "TIME_FROM_INTERNAL_OSC",
    "time": 376.510445056,
    "time_options": {
        "internal_osc": 376,
        "ptp_1588": 1651197953,
        "sync_pulse_in": 1
    }
}
}
```

- **Description:** Returns JSON-formatted sensor timing configuration and status of udp `timestamp`, `sync_pulse_in`, and `multipurpose_io`. For more information on these parameters refer to the [GET /api/v1/time](#) HTTP API command.

2.4.3 GET /api/v1/time/system

To [GET](#) the operating system time status. These values relate to the sensor operating system clocks, and not clocks related to hardware timestamp data from the lidar sensor.

```
GET /api/v1/time/system HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 389
Content-Type: application/json

{
  "monotonic": 30348.898799855,
  "realtime": 1651198091.4031146,
  "tracking": {
    "frequency": -9.558,
    "last_offset": 0.0,
    "leap_status": "not synchronised",
    "ref_time_utc": 0.0,
    "reference_id": "00000000",
    "remote_host": "",
    "residual_frequency": 0.0,
    "rms_offset": 0.0,
    "root_delay": 1.0,
    "root_dispersion": 1.0,
    "skew": 0.0,
    "stratum": 0,
    "system_time_offset": 3e-09,
    "update_interval": 0.0
  }
}
```

(continues on next page)

```
}  
}
```

- **>json float monotonic:** Monotonic time of operating system. This timestamp never counts backwards and is the time since boot in seconds.
- **>json float realtime:** Time in seconds since the Unix epoch, should match wall time if synchronized with an external time source.
- **>json object tracking:** Operating system time synchronization tracking status. See [chronyc tracking documentation](#) for more information.
- **System "tracking" fields of interest:**
 - **rms_offset:** Long-term average of the offset value.
 - **system_time_offset:** Time delta (in seconds) between the estimate of the operating system time and the current true time.
 - **last_offset:** Estimated local offset on the last clock update.
 - **ref_time_utc:** UTC Time at which the last measurement from the reference source was processed.
 - **remote_host:** This is either `ptp` if the system is synchronizing to a *PTP* time source or the address of a remote NTP server the system has selected if the sensor is connected to the Internet.

2.4.4 GET /api/v1/time/ptp

To **GET** the status of the *PTP* time synchronization daemon.

Note: See the [IEEE 1588-2008 standard for more details](#) on the standard management messages.

```
GET /api/v1/time/ptp HTTP/1.1  
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK  
Connection: keep-alive  
Content-Length: 1287  
Content-Type: application/json  
  
{  
  "current_data_set": {  
    "mean_path_delay": 0.0,  
    "offset_from_master": 0.0,  
    "steps_removed": 0  
  },  
  "parent_data_set": {  
    "gm_clock_accuracy": 254,  

```

(continues on next page)

```

"gm_clock_class": 255,
"gm_offset_scaled_log_variance": 65535,
"grandmaster_identity": "bc0fa7.ffff.008417",
"grandmaster_priority1": 128,
"grandmaster_priority2": 128,
"observed_parent_clock_phase_change_rate": 2147483647,
"observed_parent_offset_scaled_log_variance": 65535,
"parent_port_identity": "bc0fa7.ffff.008417-0",
"parent_stats": 0
},
"port_data_set": {
  "announce_receipt_timeout": 3,
  "delay_mechanism": 1,
  "log_announce_interval": 1,
  "log_min_delay_req_interval": 0,
  "log_min_pdelay_req_interval": 0,
  "log_sync_interval": 0,
  "peer_mean_path_delay": 0,
  "port_identity": "bc0fa7.ffff.008417-1",
  "port_state": "LISTENING",
  "version_number": 2
},
"profile": "default",
"time_properties_data_set": {
  "current_utc_offset": 37,
  "current_utc_offset_valid": 0,
  "frequency_traceable": 0,
  "leap59": 0,
  "leap61": 0,
  "ptp_timescale": 1,
  "time_source": 160,
  "time_traceable": 0
},
"time_status_np": {
  "cumulative_scaled_rate_offset": 0.0,
  "gm_identity": "bc0fa7.ffff.008417",
  "gm_present": false,
  "gm_time_base_indicator": 0,
  "ingress_time": 0,
  "last_gm_phase_change": "0x0000'0000000000000000.0000",
  "master_offset": 0,
  "scaled_last_gm_phase_change": 0
}
}

```

- **>json object current_data_set:** Result of the PMC `GET CURRENT_DATA_SET` command.
- **>json object parent_data_set:** Result of the PMC `GET PARENT_DATA_SET` command.
- **>json object port_data_set:** Result of the PMC `GET PORT_DATA_SET` command.
- **>json object time_properties_data_set:** Result of the PMC `GET TIME_PROPERTIES_DATA_SET` command.

- **>json object time_status_np:** Result of the PMC `GET TIME_STATUS_NP` command. This is a linux-ntp non-portable command.

Fields of interest:

- **current_data_set.offset_from_master:** Offset from master time source in nanoseconds as calculated during the last update from master.
- **parent_data_set.grandmaster_identity:** This should match the local grandmaster clock. If this displays the sensor's clock identity (derived from Ethernet hardware address) then this indicates the sensor is not properly synchronized to a grandmaster.
- **parent_data_set:** Various information about the selected master clock.
- **port_data_set.port_state:** This value will be `SLAVE` when a remote master clock is selected. See `parent_data_set` for selected master clock.
- **port_data_set:** Local sensor *PTP* configuration values. Grandmaster clock needs to match these for proper time synchronization.
- **time_properties_data_set:** *PTP* properties as given by master clock.
- **time_status_np.gm_identity:** Selected grandmaster clock identity.
- **time_status_np.gm_present:** True when grandmaster has been detected. This may stay true even if grandmaster goes off-line. Use `port_data_set.port_state` to determine up-to-date synchronization status. When this is false then the local clock is selected.
- **time_status_np.ingress_time:** Indicates when the last *PTP* message was received. Units are in nanoseconds.
- **time_status_np:** Linux *PTP* specific diagnostic values. The [Red Hat manual](#) provides some more information on these fields

2.4.5 GET /api/v1/time/ptp/profile

To `GET` the active PTP profile of the Ouster sensor.

```
GET /api/v1/time/ptp/profile HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 9
Content-Type: application/json

"default"
```

- **>json string:** Active PTP profile.

2.4.6 PUT /api/v1/time/ptp/profile

To change the PTP profile of the Ouster sensor.

```
PUT /api/v1/time/ptp/profile HTTP/1.1
Content-Type: application/json
Host: 192.0.2.123

"gptp"
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 6
Content-Type: application/json
Date: Thu, 28 Apr 2022 18:11:36 GMT
Server: nginx

"gptp"
```

<json string: PTP profile to be activated, valid options are "", "default", "gptp", "automotive-slave" and "default-l2-relaxed".

Note: "" and "default" are the same.

2.5 Alerts, Diagnostics and Telemetry

In order to correlate the alerts with sensor telemetry data, the realtime attribute that is returned in the HTTP API `/api/v1/time/system` call should be the time reference to use in such tools when timestamping telemetry data, since that will always be the time that is used by the sensor to timestamp the alerts, irrespective of the actual timing source, such as: `TIME_FROM_INTERNAL_OSC`, `TIME_FROM_SYNC_PULSE_IN`, or `TIME_FROM_PTP_1588`. This should be sufficient for tools and applications that do not require the time to have an absolute nature, i.e. where only relative time is needed.

2.5.1 GET /api/v1/sensor/alerts

To `GET` the sensor alerts.

```
GET /api/v1/sensor/alerts HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 1983
Content-Type: application/json

{
  "active": [
    {
      "active": true,
```

(continues on next page)

```

    "category": "UDP_TRANSMISSION",
    "cursor": 1,
    "id": "0x01000015",
    "level": "WARNING",
    "msg": "Client machine announced it is not reachable on the provided lidar data port;
           check that udp_dest and udp_port_lidar configured on the sensor matches
           client IP and port. This Alert may occur on sensor startup if the
           client is not listening at that time.",
    "msg_verbose": "Failed to send lidar UDP data to destination host 169.254.225.4:7502",
    "realtime": "1651197616274601728"
  },
  {
    "active": true,
    "category": "UDP_TRANSMISSION",
    "cursor": 0,
    "id": "0x01000018",
    "level": "WARNING",
    "msg": "Client machine announced it is not reachable on the provided IMU data port;
           check that udp_dest and udp_port_imu configured on the sensor matches
           client IP and port. This Alert may occur on sensor startup if the
           client is not listening at that time.",
    "msg_verbose": "Failed to send imu UDP data to destination host 169.254.225.4:7503",
    "realtime": "1651197615284695040"
  }
],
"log": [
  {
    "active": true,
    "category": "UDP_TRANSMISSION",
    "cursor": 0,
    "id": "0x01000018",
    "level": "WARNING",
    "msg": "Client machine announced it is not reachable on the provided IMU data port;
           check that udp_dest and udp_port_imu configured on the sensor matches
           client IP and port. This Alert may occur on sensor startup if the
           client is not listening at that time.",
    "msg_verbose": "Failed to send imu UDP data to destination host 169.254.225.4:7503",
    "realtime": "1651197615284695040"
  },
  {
    "active": true,
    "category": "UDP_TRANSMISSION",
    "cursor": 1,
    "id": "0x01000015",
    "level": "WARNING",
    "msg": "Client machine announced it is not reachable on the provided lidar data port;
           check that udp_dest and udp_port_lidar configured on the sensor matches
           client IP and port. This Alert may occur on sensor startup if the
           client is not listening at that time.",
    "msg_verbose": "Failed to send lidar UDP data to destination host 169.254.225.4:7502",
    "realtime": "1651197616274601728"
  }
],
"next_cursor": 2

```

}

Description: Returns JSON-formatted sensor diagnostic information.

Two lists will be returned, an **active** list and a **log** list. The **active** list contains a list of currently active events. The number of events in the active event list is unlimited.

The **log** list will contain all alert trigger and clear events. An alert-clear event has the same attributes and values as its corresponding trigger event, apart from the realtime and **cursor** attributes which will have increased, since an alert-clear event will always be received after an alert-trigger event. The **log** list has a length limit of 32 events in the form of a FIFO (First in First Out) queue. When the **log** list length limit is reached and a new event is added the oldest event is deleted.

In addition to the **active** and **log** lists, `GET /api/v1/sensor/alerts` also returns a **next_cursor** field. Every alert event has a **cursor** attribute, which increments for every alert event logged. This can be used to track the alert activity that has been viewed and reduce message bandwidth. To do this, users are recommended to save the **next_cursor** field when calling `GET /api/v1/sensor/alerts` and then use that value as the **cursor** argument on the next `GET /api/v1/sensor/alerts` call to fetch only new **log** entries.

A valid value for **mode** is either `summary` or `default`.

Additional Information:

The **cursor** will wrap at 2^{32} entries. It is important to understand the behavior during the wrap case, since this may lead to some unexpected consequences:

- If $\text{cursor} < (\text{next_cursor} - 32) \% 2^{32}$ then some entries may be filtered. For instance if **cursor** = 0 and **next_cursor** = 0 no entries will be reported immediately after **cursor** wrap, even though the log contains 32 entries, where submitting **cursor** = 4294967264 $(\text{next_cursor} - 32) \% 2^{32}$ will return all logged values.
- If $\text{cursor} > \text{next_cursor}$ all 32 entries will be reported.

The recommended approach to using the interface is to always base queries on the previous value of **next_cursor**.

Alerts Example

Valid uses of `GET /api/v1/sensor/alerts`:

- Example: Calling alerts with **cursor** = 1
 - `GET /api/v1/sensor/alerts?cursor=1`
- Example: Calling alerts with **mode** = summary
 - `GET /api/v1/sensor/alerts?mode=summary`
- Example: Calling alerts with **cursor** = 2 and **mode** = summary

- *GET /api/v1/sensor/alerts?cursor=2&mode=summary*

Note: When utilizing HTTP Endpoints, the sequence in which the cursor and mode arguments are provided for the `GET /api/v1/sensor/alerts` command is inconsequential.

The alerts reported have the following format:

```
{
  "category": "Category of the alert: e.g. OVERTEMP, UDP_TRANSMISSION",
  "level": "Level of alert: e.g. NOTICE, WARNING, ERROR",
  "realtime": "The timestamp of the alert in nanoseconds",
  "active": "Whether the alert is active or not: <true/false>",
  "msg": "A description of the alert",
  "cursor": "The sequential number of the alert, starting from 0 counting up",
  "id": "The hexadecimal identification code of the alert: e.g. 0x01000017",
  "msg_verbose": "Any additional verbose description that the alert may present"
}
```

2.5.2 GET /api/v1/sensor/alerts?cursor=1

To `GET` the sensor alerts with `cursor=1`.

```
GET /api/v1/sensor/alerts?cursor=1 HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 43
Content-Type: application/json
Date: Thu, 21 Mar 2024 00:01:55 GMT
Server: nginx
```

```
{
  "active": [],
  "log": [],
  "next_cursor": 0
}
```

2.5.3 GET /api/v1/sensor/alerts?mode=summary

To `GET` the sensor alerts with `mode=summary`.

```
GET /api/v1/sensor/alerts?mode=summary HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 351
Content-Type: application/json
Date: Thu, 21 Mar 2024 00:11:50 GMT
Server: nginx
```

(continues on next page)

(continued from previous page)

```
{
  "active": [
    {
      "cursor": 1,
      "id": "0x01000015",
      "realtime": "1710979334555586816"
    },
    {
      "cursor": 0,
      "id": "0x01000018",
      "realtime": "1710979333553802752"
    }
  ],
  "log": [
    {
      "active": true,
      "cursor": 0,
      "id": "0x01000018",
      "realtime": "1710979333553802752"
    },
    {
      "active": true,
      "cursor": 1,
      "id": "0x01000015",
      "realtime": "1710979334555586816"
    }
  ],
  "next_cursor": 2
}
```

2.5.4 GET /api/v1/sensor/alerts?cursor=2&mode=summary

To **GET** the sensor alerts with cursor=2 and mode=summary.

```
GET /api/v1/sensor/alerts?cursor=2&mode=summary HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 1011
Content-Type: application/json
Date: Thu, 21 Mar 2024 00:21:34 GMT
Server: nginx
```

```
{
  "active": [
    {
      "active": true,
      "category": "UDP_TRANSMISSION",
      "cursor": 1,
      "id": "0x01000015",
```

(continues on next page)

(continued from previous page)

```
    "level": "WARNING",
    "msg": "Client machine announced it is not reachable on the provided lidar data port;
check that udp_dest and udp_port_lidar configured on the sensor matches client IP and port.
This Alert may occur on sensor startup if the client is not listening at that time.",
    "msg_verbose": "Failed to send lidar UDP data to destination host 10.32.224.28:7502",
    "realtime": "1710979334555586816"
  },
  {
    "active": true,
    "category": "UDP_TRANSMISSION",
    "cursor": 0,
    "id": "0x01000018",
    "level": "WARNING",
    "msg": "Client machine announced it is not reachable on the provided IMU data port;
check that udp_dest and udp_port_imu configured on the sensor matches client IP and port.
This Alert may occur on sensor startup if the client is not listening at that time.",
    "msg_verbose": "Failed to send imu UDP data to destination host 10.32.224.28:7503",
    "realtime": "1710979333553802752"
  }
],
"log": [],
"next_cursor": 2
}
```

2.5.5 GET /api/v1/diagnostics/dump

To **GET** the diagnostics files of the sensor. This file should be sent to Ouster support if requested, and is not readable by the user.

```
GET /api/v1/diagnostics/dump HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Type: application/octet-stream
Transfer-Encoding: chunked
content-disposition: attachment; filename="192.0.2.123_diagnostics-dump_b7d348c2
-c763-11ec-accf-bc0fa7008417.bin"

+-----+
| NOTE: binary data not shown in terminal |
+-----+
```

2.5.6 GET /api/v1/sensor/telemetry

To **GET** the sensor telemetry information.

```
GET /api/v1/sensor/telemetry HTTP/1.1
Host: 192.0.2.123
```

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 150
Content-Type: application/json

{
  "input_current_ma": 644,
  "input_voltage_mv": 23624,
  "internal_temperature_deg_c": 48,
  "phase_lock_status": "DISABLED",
  "timestamp_ns": 2093396806056
}
```

Description: Returns a JSON-formatted response that provides sensor system state information. This includes the **Timestamp** in **ns** (Nanoseconds) at which the information was collected, **Lidar Input Voltage** in **mv** (Millivolt), **Lidar Input Current** in **ma** (Milliamp), **Internal Temperature** of the sensor in **°C** (Degree Celsius) and **Phase Lock status**, which can be **LOCKED**, **LOST**, or **DISABLED**..

Internal temperature can only be measured with Rev 06 and above sensors. **Phase lock** output will not indicate loss of lock if the time synchronization is lost.

3 TCP API Guide (Deprecated)

Warning: TCP API has now been deprecated in FW 3.1. Refer to HTTP API Guide Section instead.

Please contact [Ouster Support](#) if you need any support or have any questions regarding this transition.

4 API Changelog

4.1 Firmware v3.1.0

- Date: May 2024

Added

- Add config parameter for `min_range_threshold_cm` (Refer to [min_range_threshold_cm](#) for more information).
- Add config parameter for `return_order` (Refer to [return_order](#) for more information).
- Add config parameter for `Delete Config` (Refer to [DELETE /api/v1/sensor/config](#) for more information).
- Add config parameter for [POST /api/v1/system/restart](#) to restart sensor or reinitialize a sensor.
- Add config parameter for [User Editable Data](#) section, which can be used for a number of purposes such as storing specific information about the sensor, qualifying a sensor, calibration data, or any other information.
- Add config parameter for [GET /api/v1/sensor/metadata/imu_data_format](#). User can get `imu_data_format` and [POST](#) config to change `gyro_fsr` and `accel_fsr` from `NORMAL` to `EXTENDED`.

Removed

- TCP API has now been **DEPRECATED** in FW 3.1. Please refer to [HTTP API Reference Guide](#) section instead.
- LEGACY Data packet profile has been **DEPRECATED**, please refer to Lidar Data Packet Format section of the Firmware User Manual for more information.

Fixed

- Fixed bug in [GET /api/v1/sensor/alerts](#) API.

4.2 Firmware v3.0.1

- Date: February 2023

Added

- New HTTP Command to configure speed override (Refer to [System](#))

4.3 Firmware v3.0.0

- Date: January 2023

Fixed

- Bug in keep-alive behavior for HTTP 1.1.