

# Socket Manual

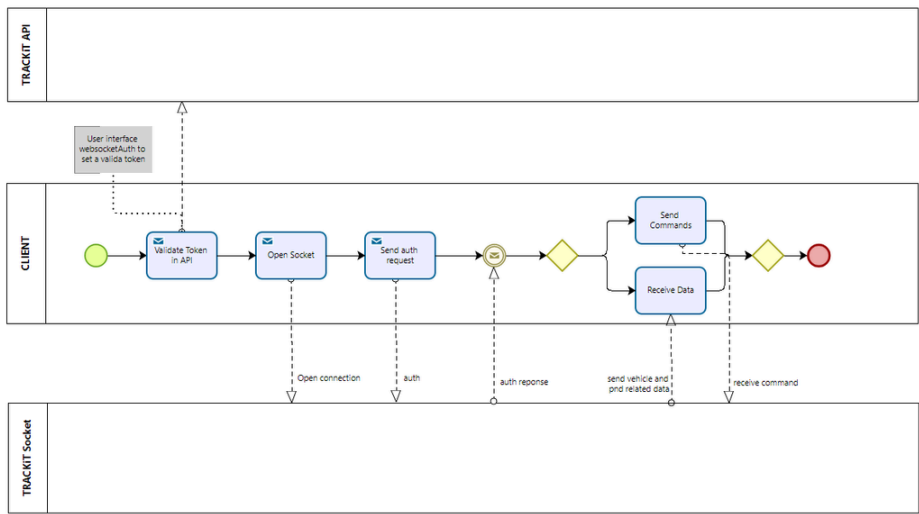
Sockets (websockets) are used to receive data from devices (tracking and TRACKiT Works) in real time.

**Socket:** wss://websocket.trackit.pt

**API:** <https://i.trackit.pt/ws/>

## This is the process to authenticate and use the socket: [🔗](#)

1. Authenticate in the main API
2. Validate the authentication token
3. Connect to the TRACKiT socket
4. Send an auth request



Process of socket authentication

## Authentication token [🔗](#)

A valid token is required to authenticate on the socket.

The token can be retrieved using the main API, with the interface [websocketAuth](#). Documentation [here](#).

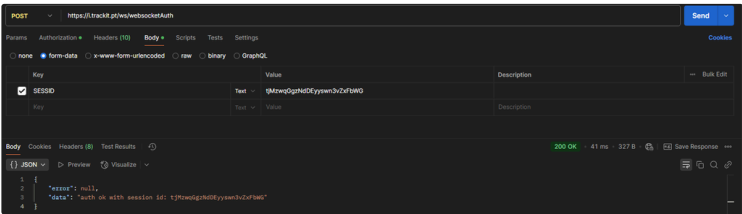
## HTTP Request [🔗](#)

POST <https://i.trackit.pt/ws/websocketAuth>

### Parameters [🔗](#)

- SESSID
  - Description: Token validation string
  - Type: string
  - Length: 26

### Request example: [🔗](#)



Request example using Postman

```
1 var request = require('request');
2 var options = {
3   'method': 'POST',
4   'url': 'https://i.trackit.pt/ws/websocketAuth',
5   'headers': {
```

```
6   'Authorization': '.....'
7 },
8   formData: {
9     'SESSID': 'tjMzwqGgzNdDEyyswn3vZxFbWG'
10  }
11 };
12 request(options, function (error, response) {
13   if (error) throw new Error(error);
14   console.log(response.body);
15 });
```

## Command list [↗](#)

Here is the list of all commands that can be sent to the socket

### Authentication [↗](#)

Authenticate on the socket

**Command:** auth

| Query  | Description                | Example   | Response   |
|--------|----------------------------|---|--|
| (none) | Authenticate on the socket | <pre>1 { 2   "type": "auth", 3   "query": { 4     "user_id": 123456, 5     "SESSID": "tjMzwqGgzNdDEyyswn3vZxFbWG" 6   } 7 }</pre> | <pre>1 { 2   "params": { 3     "type": "auth", 4     "success": true 5   } 6 }</pre> |

### API Command [↗](#)

Send commands to the tracking device

**Command:** api

| Query | Description                              | Example  | Response  |
|-------|--|--|---|
| getio | Get the current status of the device I/O | <pre>1 { 2   "type": "api", 3   "query": { 4     "id": "ZS8SYGKBXAI40T2ASQ3R3HW95", 5     "uid": 1001, 6     "cmd": "getio", 7     "CR": false 8   } 9 }</pre> | <pre>1 { 2   "uid": 1001, 3   "params": { 4     "type": "cmd-status", 5     "id": "ZS8SYGKBXAI40T2ASQ3R3HW95", 6     "status": true 7   } 8 }</pre> |

### PND API Command (pnd-api) [↗](#)

Send commands to the driver's app device (TRACKIT Works)

**Command:** pnd-api

| Query | Description                            | Example   | Response  |
|-------|--|---|---|
| msg   | Will send a text message to the driver | <pre>1 { 2   "type": "pnd-api", 3   "query": { 4     "uid": 1001, 5     "content": { 6       "type": "msg", 7       "txt": "this is my message", 8       "user_id": 123456, 9       "user_name": "John Doe", 10      "guid": "ZS8SYGKBXAI40T2ASQ3R3HW95" 11     } 12   } 13 } 14 } 15 }</pre> | <pre>1 { 2   "uid": 1001, 3   "params": { 4     "type": "msg-sts", 5     "id": "67ed11c70191a1b1c70191a1b1c70191", 6     "txs": 0, 7     "content": { 8       "type": "msg", 9       "content": { 10        "txt": "this is my message", 11        "user_id": 123456, 12        "user_name": "John Doe", 13        "guid": "ZS8SYGKBXAI40T2ASQ3R3HW95" 14      } 15     } 16   } 17 }</pre> |

## Data Parameters ID

Here is the list of data types that are sent by the socket

### Vehicle-related data

Data received from vehicles, like positioning and events

| Property | Description  | type   | Example   |
|----------|--|--------|---|
| records  | <p>Receive the last record sent in a batch by the device.</p> <p>Ex: If the device uploads 10 records, the socket will send only the last one and the information about the number of records received.</p> <p>For documentation about the lastRecord content, and a list of the different properties, can be found in the <a href="#">API manual</a>, <b>vehicleHistoric</b> interface.</p>   | object | <pre>1 { 2   "uid": 1001, 3   "params": { 4     "type": "records", 5     "total": 1, 6     "lastRecord": { 7       "tmx": "2022-04-02T09:50:03.574Z", 8       "vid": 826403, 9       "dmx": 1, 10      "loc": { 11        "lon": -9.0127133, 12        "lat": 38.5787133 13      }, 14      "gsp": 41, 15      "hdg": 246, 16      "gkm": 1613709, 17      "cas": 3, 18      "exd": { 19        "sat": 15, 20        "alt": 42, 21        "bpp": 0, 22        "eco": { 23          "edt": 1, 24          "aat": 11, 25          "rpm": 949, 26          "ehr": 25237, 27          "flv": 100, 28          "dcl": "0000001993887010" 29        }, 30        "io": { 31          "1": 1, 32          "8": 1 33        } 34      } 35    } 36  } 37 }</pre>  |
| event    | <p>All I/O Events are sent via the socket (<i>sub-type: io</i>).</p> <p>Some other events are also sent (<i>sub-type: event</i>):</p> <ul style="list-style-type: none"><li>The device starts using the internal battery</li><li>External power provided</li><li>Movement detection</li><li>Movement detection without driver identification</li></ul> <p>For documentation about the <b>record</b> content and a list of the different properties, can be found in the <a href="#">API manual</a>, <b>vehicleHistoric</b> interface.</p> <p>For documentation about the <b>event</b> content and a list of the different properties, can be found in the <a href="#">API manual</a>, <b>events</b> interface.</p> | object | <pre>1 { 2   "uid": 1001, 3   "params": { 4     "type": "event", 5     "sub-type": "io", 6     "record": { 7       "_id": "67ed0c4f0191a78e5aa92266", 8       "tmx": "2022-04-02T10:07:03.000Z", 9       "vid": 823397, 10      "dmx": 8, 11      "loc": { 12        "lon": 1.20606, 13        "lat": 41.2165983 14      }, 15      "gsp": 0, 16      "hdg": 297, 17      "gkm": 398447, 18      "cas": 61, 19      "exd": { 20        "sat": 11, 21        "tlm": { 22          "tio": 1, 23          "tlv": 0 24        } 25      } 26    }, 27     "event": { 28       "_id": "67ed0c4f0191a78e5aa92269", 29       "hid": "67ed0c4f0191a78e5aa92266", 30       "mid": 1001, 31       "eid": 3846, 32       "tmx": "2022-04-02T10:07:03.000Z", 33       "rmx": "2022-04-02T10:07:11.398Z", 34       "lat": 41.2165983, 35       "lng": 1.20606,</pre> |

|            |  |        |   |
|------------|--|--------|---|
|            |  |        | <pre> 36         "sts": 1, 37         "frc": "(...), Espanha", 38         "src": "tlk-server" 39     } 40 } 41 }</pre>  |
| connection | If a new device is connected/disconnected to the server  | object | <pre> 1 { 2     "uid": 1001, 3     "params": { 4         "type": "connection", 5         "status": 1 6     } 7 }</pre>  |
| cmd-parsed | This will query the device directly, so it must be used by skilled users who know what they are doing. For that reason, this command is limited. | mixed  | <pre> 1 { 2     "uid": 1001, 3     "params": { 4         "type": "cmd-parsed", 5         "sub-type": "getio", 6         "cmd": { 7             "1": { 8                 "tio": 1, 9                 "tlv": 0, 10                "hid": null, 11                "tmx": "2022-04-02T10:06:59.321Z" 12            }, 13            "2": { 14                "tio": 2, 15                "tlv": 0, 16                "hid": null, 17                "tmx": "2022-04-02T10:06:59.321Z" 18            } 19        } 20     } 21 }</pre> |

## TRACKiT Works related data [🔗](#)

Data received from TRACKiT Works, like events, forms, messages, etc.

| Property       | Description  | type    | Example   |
|----------------|--|---------|---|
| pnd-connection | If a new device is connected/disconnected to the server  | boolean | <pre> 1 { 2     "uid": 1001, 3     "params": { 4         "type": "pnd-connection", 5         "status": 1 6     } 7 }</pre>  |
| pnd-msg        | Text message sent by the driver.<br><br>For documentation about the <b>event</b> content and a list of the different properties, can be found in the <a href="#">API manual</a> , <b>events</b> interface.             | object  | <pre> 1 { 2     "uid": 1001, 3     "params": { 4         "type": "pnd-msg", 5         "event": { 6             "hid": null, 7             "mid": 1001, 8             "tmx": "2022-04-02T10:24:42.000Z", 9             "rmx": "2022-04-02T10:24:42.002Z", 10            "lat": 41.2165983, 11            "lng": 1.20606, 12            "sts": 1, 13            "frc": null, 14            "src": "tlk-server", 15            "eid": 3951, 16            "txt": "text message sent by driver" 17        } 18     } 19 }</pre> |
| pnd-msg-prd    | Pre-defined text message sent by the driver.<br><br>For documentation about the <b>event</b> content and a list of the different properties, can be found in the <a href="#">API manual</a> , <b>events</b> interface. | object  | <pre> 1 { 2     "uid": 1001, 3     "params": { 4         "type": "pnd-msg-prd", 5         "event": { 6             "hid": null, 7             "mid": 1001, 8             "tmx": "2022-04-02T10:24:42.000Z", 9             "rmx": "2022-04-02T10:24:42.002Z",</pre>  |

|                 |   |        |  |
|-----------------|---|--------|--|
|                 |   |        | <pre> 10         "lat": 41.2165983, 11         "lng": 1.20606, 12         "sts": 1, 13         "frc": null, 14         "src": "tlk-server", 15         "eid": 25106, 16         "pmi": 1, 17         "txt": "TRAFFIC" 18     } 19 } 20 }</pre>   |
| form            | <p>Form filled and sent by the driver.</p> <p>For documentation about the <b>event</b> content and a list of the different properties, can be found in the <a href="#">API manual</a>, <b>events</b> interface.</p>   | object | <pre> 1 { 2     "uid": 1001, 3     "params": { 4         "type": "form", 5         "event": { 6             "hid": null, 7             "mid": 1001, 8             "tmx": "2022-04-02T10:27:00.000Z", 9             "rmx": "2022-04-02T10:27:34.223Z", 10            "lat": 41.2165983, 11            "lng": 1.20606, 12            "sts": 1, 13            "frc": null, 14            "src": "tlk-server", 15            "eid": 33, 16            "frm": { 17                "fnm": "Form name", 18                "fld": { 19                    "Client": "(some text)", 20                    "Work": "(some text)", 21                    "Local": "(some text)", 22                    "...": "(some value)" 23                } 24            } 25        } 26    } 27 }</pre>   |
| msg-sts         | <p>This is the response when you send a text message to a driver. The field txs can have the following values:</p> <ul style="list-style-type: none"> <li>• 0: Not processed</li> <li>• 1: Processed (buffered)</li> <li>• 2: Not sent (error)</li> <li>• 3: Pending</li> <li>• 4: Sent with confirmation</li> <li>• 5: Message read</li> </ul> | object | <pre> 1 { 2     "uid": 1001, 3     "params": { 4         "type": "msg-sts", 5         "id": "67ed11c70191a78e5aaae7d7", 6         "txs": 1 7     } 8 }</pre>   |
| service-generic | <p>Receive information about the services/mission activity of the driver, like whether the service was delivered or visited.</p>  | object | <pre> 1 { 2     "uid": 1001, 3     "params": { 4         "type": "service-generic", 5         "event": { 6             "tmx": "2022-04-02T05:47:07.000Z", 7             "vid": 337145, 8             "dmx": 0, 9             "loc": { 10                 "lat": 41.2165983, 11                 "lng": 1.20606, 12             }, 13             "gsp": 0, 14             "hdg": 0, 15             "gkm": -1, 16             "cas": 600, 17             "exd": { 18                 "type": "service-generic", 19                 "evt": { 20                     "id": "67ec4295af04705c2e0000", 21                     "delivered": true, 22                     "tmx": "2022-04-02T06:47:07.0", 23                     "msg": "Delivered" 24                 } 25             }, 26             "_id": "67eccf5c4219b1b25e0aca0d" 27         } 28     } 29 }</pre> |

