

User's manual

Connection ZSL devices to SENT GEO

Material made available to the public

Warsaw 6.05.2019

Table of Contents

1	Introduction.....	3
2	Registration of the location data transmission service by the Operator of the ZSL on behalf of (in favor of) the Carrier in SENT on PUESC (SENT 460).....	5
3	Information provided by the ZSL Operator to the Carrier	6
4	Registration by the Carrier GPS devices used to register the current location of carriage in SENT on PUESC (SENT 470):.....	7
5	Transmission by the Carrier of the ZSL Operator of the list of the technical identifiers of registered location devices.....	7
6	Transmission of location data by the ZSL Operator to SENT GEO from devices indicated by the Carrier.....	8
7	Sgdi_rest_request_schema_v_0_5.json.....	8
8	Sgdi_rest_request_schema_v_0_6.json.....	11
9	Sgdi_rest_request_schema_v_0_61.json.....	13
10	Method of data transfer.....	16
11	Security of transmitted data	16
12	Validation of data – duties on the part of ZSL Operator	16
13	Data validation – list of messages	16
14	Information necessary to connect the ZSL to SENT GEO	20
15	Application of certificates.....	20
16	Tests.....	30
17	Contact	31
18	Good advice	31
19	Interface for downloading the last location by ZSL operators	32

1 Introduction

Target process of production geolocation data delivery to production SENT GEO module data interface will covers following phases:

- External Localisation System (ELS) service Operator (or simple ELS Operator or in polish language ZSL) register yourself on PUESC (for production environment <https://puesc.gov.pl> and for testing environment <https://test.puesc.gov.pl/>),
- registration of geolocation data delivery service by ELS service Operator in SENT on PUESC with SENT 460 XML document or SENT 460 form,
- registered ZSL service Operator provides to Carrier information necessary to start the process of data transfer.
- Carrier register his geolocation devices (which will be used to provide geolocation data to SENT GEO module) in SENT on PUESC (SENT 470):
- Carrier provides registered geolocation technical device identifier to registered ZSL service Operator – this lets registered ZSL service Operator to know which geolocation data of which geolocation devices should be delivered to SENT GEO module,
- registered ZSL service Operator starts to delivery geolocation data for Carrier selected geolocation devices to SENT GEO module data interface.

Collecting location data by ZSL operators should occur each time one of the following criteria occurs:

- 1) **maximum after traveling 1 km or after 1 minute** depending on which event occurs later, but not less than every 5 minutes;
- 2) **when the azimuth of the vehicle's direction is changed by an amount equal to or greater than 40 degrees;**
- 3) **at a standstill of the means of transport, with the engine running, every 5 minutes;**
- 4) **before turning off the engine**, indicating where the vehicle stopped for a longer break resulting from regulations related to the driver's working time and **after switching on the engine**, thus indicating the end of the break.

Location data meeting the above conditions should be transferred from the location device to the SENT GEO module with a delay of no more than 1 minute. Location data collected by the location device can be buffered by ZSL operators, and then in a cycle of not more than 1 minute sent in packets to the SENT GEO module, with a restriction on the size, which is 500 KB and the number of minimum positions 1, and the maximum of 500 Data that exceeds 500 KB and / or 500 items should be divided into the appropriate number of packages. The data within the package can come from different devices and meet the various criteria defined above.

Examples to criterion 1:

- 1) 30 km / h - later there is a criterion of 1 km - 1 km traveled, sample sent after 2 minutes,

- 2) 60 km / h - criterion of 1 km and 1 min occurs at the same moment - 1 km traveled, sample sent after 1 min,
- 3) 90 km / h - later there is a criterion of 1 minute - 1.5 km traveled, sample sent after 1 minute,
- 4) 120 km / h - later there is a criterion of 1 minute - 2 km traveled, sample sent after 1 min.

Until 31 December 2018, the following applies: scheme JSON

`sgdi_rest_request_schema_v_0_5.json`.

However, **from 1 January 2019**, the following scheme applies: JSON

`sgdi_rest_request_schema_v_0_6.json`.

Failure to comply with the above rules will result in **the carriers failing to fulfill their obligations under Art. 10a of the Act on the system of monitoring road and rail transport of goods.**

The transmitted data will be verified in terms of meeting the above criteria. If they are found to be exceeded, the ZSL Operators will be periodically informed of the irregularities found to remove them.

Since January 1st, we kindly remind that the new schema (`sgdi_rest_request_schema_v_0_6.json`) took effect to validate data transmitted through External Localization System (ZSL). In the schema, below fields are required:

1. lat (latitude)
2. lon (longitude)
3. tsp (date and time of a sample)
4. dev (technical ID of a device)
5. brg (azimuth/bearing in degrees)
6. acc (accuracy in meters)
7. spd (speed in meters per second)

Since February 1st, transmitting data incompatible to the schema `sgdi_rest_request_schema_v_0_6.json` will have the effect of rejection the transmitted data.

We kindly please to immediate adjustment to implemented amendments. Attention should be paid to often mistakes in transmitting data:

1. 'dev' filed – transmitting device number (Znn-xxnnxx-n) instead of device technical ID.
2. 'dev' field – transmitting different device number than the registered through SENT_47x (e.g. case sensitive, adding prefix or suffix).
3. 'lat' field - transmitting data outside of Poland. Data which fall outside the scope of 49.0 - 54.835778 should not be transmitted.
4. 'lon' field – transmitting data outside of Poland. Data which fall outside the scope of 14.116667 - 24.15 should not be transmitted.
5. 'tsp' field - transmitting data from the future which are the result of incorrect implementation of a time zone (timestamp shall specify the UTC time).

6. 'lat' & 'lon' field – too high accuracy. 10 decimal places are permitted.
7. 'spd' field – data in this field should be transmitted in meters per second.
8. 'id' field- transmitting static value in 'id' field. This field is a record ID so it should contains unique data. It could be a key field of your gps data table. This field is not required.

In case of 1) and 2) transmitting incorrect data does not give an error feedback as a result, according to the findings, External Localization System (ZSL) should transmit data of all vehicles, even those not registered. Since February 1st transmitting that kind of data results in add a warning to a feedback. In case of 2), in order to minimize number of mistakes during transmitting data, since February 1st, case sensitive will not be considered in 'dev' field. In case of 3), 4), 6) and 7) transmitting that kind of data result in error 400. If a data package will contain more than one record, occurrence of at least one incorrect record results in rejection of the whole package and need of renewed transmission of the package without incorrect records. In case of 5) data will be accepted. Since February 1st that kind of data result in a feedback which contains a warning. In future that kind of data will be rejected. In case of 8) data will be accepted.

In order to validation of data transmission, we kindly please to use the test interface. Test interface can be registered through <https://test.puesc.gov.pl/> page. Currently, test interface operates in accordance with the schema sgdi_rest_request_schema_v_0_6.json and additional warnings are added to a feedback.

2 Registration of the location data transmission service by the Operator of the ZSL on behalf of (in favor of) the Carrier in SENT on PUESC (SENT 460)

This phase consists of the following steps:

- a. Operator provides:
 - i. server's IPv4 addresses of these servers which will provide geolocation data delivery to SENT GEO module within registered ZSL service,
 - ii. SSL/TLS certificate request for clients (one SSL/TLS client certificate for all) which will provide geolocation data delivery to SENT GEO module,
 - iii. contact data (e-mail address and phone number) which will be used by Operator server's administrator for technical help or contact,
- b. Operator receives:
 - i. the number of the ZSL service for registered geolocation data delivery ZSL service (registered service, registered ZSL service),
 - ii. authorization password assigned to the ZSL service for operations performed by the Service Provider,

- iii. Carrier start password for Carrier which lets Carrier to register his geolocation devices (devices which delivery these geolocation data to Operator ZSL and next via Operator ZSL to SENT GEO module), Operator have to provide Carrier authentication password to Carrier,
- iv. service related SENT GEO module data interface URL address provided by SENT GEO module exclusively for registered geolocation data delivery service, ZSL will send geolocation data to this SENT GEO module data interface URL address within registered service,
- v. client SSL/TLS certificate which has been used by ZSL Operator for data delivery authentication;

Note I: ZSL Operator becomes registered ZSL service Operator.

Note III: ZSL Operator register one geolocation data delivery service for one client (Carrier), or one service to support many Carriers. The ZSL operator can register any number of ZSL services.

Note III: If it is in need (e.g. client certificate has been lost, client certificate password has been compromised, client certificate expired, and so on), client certificate could be renewed with new SSL/TL certificate request (SENT 463).

Note IV: Non-updated attributes of the registered service remain the same and are returned by PUESC in SENT 461 (Service registration confirmation message External Service Systems (ZSL) - the message is intended only for the ZSL Service Operator) despite the fact that they have not changed. Therefore, the certificates returned when adding / changing / editing IP numbers are the same as those that the operator received previously when registering the service and do not need to replace them with those returned to that SENT 461.

Note V: Information on returning the current value of all attributes of the registered ZSL service by PUESC in SENT 461 (Service registration confirmation message External Service Systems (ZSL) - message is intended only for the ZSL service operator) when updating at least one attribute using SENT 463 (Service Edit message ZSL Operator - for the needs of the SENT-GEO service).

3 Information provided by the ZSL Operator to the Carrier

The ZSL operator provides the following information to the Carrier:

- a. registered ZSL Operator the number of the ZSL service,
- b. Carrier start password which lets Carrier to register his geolocation devices;

Note: After the Carrier has registered GPS devices connected to the ZSL service (SENT 470), the SENT system generates a new authorization password for the Carrier for operations performed by the GPS device manager.

4 Registration by the Carrier GPS devices used to register the current location of carriage in SENT on PUESC (SENT 470):

Carrier register his geolocation devices (which will be used to provide geolocation data to SENT GEO module) in SENT on PUESC (SENT 470):

The registration by the Carrier of geolocation devices in SENT includes the following steps.

- a. Technical identifiers of carrier location devices related to the ZSL service (maximum of 100 items in one SENT 470 XML message or SENT 470 form). If there is a need to add more technical identifiers, the Carrier does this by means of the SENT 473 XML message or the SENT 473 form (maximum 100 technical identifiers). The Carrier can send any number of XML SENT messages 473 or SENT 473 forms.

Note: the technical identifier of the geolocation device is usually hardware id and it is stored in geolocation device memory or hardcoded in electronic chip. You could ask your ZSL service Operator about technical ids of geolocation devices mounted in your vehicles.

Note: the technical identifier of the GPS device has to be included in geolocation data delivered by ZSL service Operator to SENT GEO module.

- b. Carrier receives from SENT on PUESC geolocation GPS device number which will be used by Carrier to register SENT carriage in SENT on PUESC,

Note: One GPS device number is provided for one the technical identifier of the GPS device.

5 Transmission by the Carrier of the ZSL Operator of the list of the technical identifiers of registered location devices

Carrier provides registered the technical identifier of the GPS device to registered ZSL service Operator – this lets registered ZSL service Operator to know which geolocation data of which GPS devices should be delivered to SENT GEO module.

6 Transmission of location data by the ZSL Operator to SENT GEO from devices indicated by the Carrier

Registered ZSL service Operator starts to delivery geolocation data for Carrier selected geolocation devices (see phase 4) to SENT GEO module data interface, geolocation data are delivered with:

- a. the URL address provided by SENT during ZSL service registration (2.b.iv),
- b. HTTPS protocol and SSL/TLS client certificate authorisation, registered ZSL service Operator uses SSL/TLS client certificate provided by SENT during ZSL service registration or the renewed on (2.b.v),
- c. REST mechanism and HTTP POST method, send geolocation data as JSON data structures which are compatible with provided current JSON schema.

Before start of production geolocation data delivery to production SENT GEO module data interface, our team (SENT GEO team) provides access to test SENT GEO module data interface (test SENT GEO data interface), which will require additional steps provided by you (Carrier), your ZSL Operator and our team.

7 Sgdi_rest_request_schema_v_0_5.json

By the end of 2018, the JSON scheme (sgdi_rest_request_schema_v_0_5.json) applies. The limit on the size of the transmitted packet is 1 MB. The minimum number of vehicle data entries in the JSON table is 1, and the maximum number is of 500.

The following changes have been made to the sgdi_rest_request_schema_v_0_4.json version:

- increasing the character limit for the dev attribute (identifier of the locator) to 50,
- change of the minimum limit to -1000 and the maximum to 4000 for the optional alt attribute (height),
- adding null data support for optional attributes,
- an explicitly defined limit of 500 items in the data table,
- explicitly defined ban on sending attributes not entered in the schema,
- the limit of the package size in the amount of 1 MB entered in the comment;

The sgdi_rest_request_schema_v_0_5.json scheme does not require the presence of all attributes entered in the Act ("brg", "dev", "lat", "lon", "spd", "acc", "tsp").

The following is an example of JSON (sgdi_rest_request_schema_v_0_5.json):


```
[
  {
    "id": "1",
    "dev": "3535A058060495422C7934",
    "lat": 52.17237393,
    "lon": 21.1941930753,
    "tsp": 1539186656000000,
  },
  {
    "id": "2",
    "dev": "35A058060495422C7947",
    "lat": 51.17237393,
    "lon": 20.1941930753,
    "tsp": 1539186676000000,
  }
]
```

The data is transferred in the form of a JSON table, in which individual elements are JSON objects containing individual waypoints. The description of individual fields, the rules for validation and information about field maturity in Schema _ v_0_ 5 is presented in Table 1.

Table 1. Schema_v_0_5

Name	Description	Validation rule	Required
id	The unique identifier of the record in the source system, the variable used for verification purposes during the testing period.	"type": "string", "minLength": 1, "maxLength": 36, "examples": ["1", "1960472"]	No
dev	Unique identifier of the locator, maximum length of 50 characters allowed.	"type": "string", "minLength": 1, "maxLength": 50, "examples": ["000000000000B1", "35A058060495422C7934"]	Yes
lat	Latitude taken from the GPS transmitter, WGS reference system 84, recommended minimum number of decimal places: 6, maximum number of decimal places allowed: 10. The limits of assumed extreme coordinates: 49.0 N and 54.835778 N are adequately reduced or increased by 1 degree.	"type": "number", "minimum": 48.0, "maximum": 55.835778, "multipleOf": 0.0000000001, "examples": [52.0375868826, 52.172644]	Yes
lon	Longitude taken from the GPS transmitter, WGS reference	"type": "number", "minimum": 13.116667, "maximum": 25.15,	Yes

	system 84, recommended minimum number of decimal places: 6, maximum number of decimal places allowed: 10. The limits of the assumed outer coordinates: 14.116667 E and 24.15 E are adequately reduced or increased by 1 degree.	"multipleOf": 0.0000000001, "examples": [21.1956136, 20.026094]	
alt	Ellipsoidal height taken from the GPS transmitter, unit [m], maximum number of decimal places allowed: 2.	"type": ["number", "null"], "minimum": -1000.0, "maximum": 4000.0, "multipleOf": 0.01, "examples": [10.0, 200.02]	No
tsp	Timestamp includes date and time taken from the GPS transmitter, associated with the geographical position of the record, UTC time zone, SENT GEO timestamp is similar to Epoch / Unix Timestamp, but given to the microsecond (16 digits), this is the number of microseconds that have passed since 00:00:00 Coordinated Universal Time (UTC), Thursday, 1 January 1970, the minimum value indicates 2017.09.20 00:00:00 UTC, integer.	"type": "integer", "minimum": 1505865600000000, "examples": [1506086623000000, 1511273867317000]	Yes
spd	Movement speed taken from the GPS transmitter - unit [m/s], maximum number of decimal places allowed: 2, maximum allowed speed: 56.00 [m/s].	"type": ["number", "null"], "minimum": 0.0, "maximum": 56.0, "multipleOf": 0.01, "examples": [3.21, 20.0]	No
acc	Location accuracy taken from the GPS transmitter - circle radius in meters, maximum number of decimal places allowed: 2.	"type": ["number", "null"], "minimum": 0.0, "multipleOf": 0.01, "examples": [10.14, 30.0]	No
brg	Azimuth - unit [grade], maximum number of decimal places allowed: 2.	"type": ["number", "null"], "minimum": 0.0, "maximum": 360.0, "multipleOf": 0.01, "examples": [40.14, 230.0]	No

8 Sgdi_rest_request_schema_v_0_6.json

From January 1, 2019, the JSON scheme (sgdi_rest_request_schema_v_0_6.json) applies. The limit on the size of the transmitted packet is 500 KB. The minimum number of vehicle data entries in the JSON table is 1, and the maximum number is of 500.

The following changes have been made to the sgdi_rest_request_schema_v_0_5.json scheme:

- the requirement of all attributes entered in the Act ("brg", "dev", "lat", "lon", "spd", "acc", "tsp"),
- pattern for verification of the dev attribute (device) in the form "^ [a-zA-Z0-9 \\ _] {1,50} \$", the pattern allows to enter uppercase and lowercase letters, numbers, dash-minus characters "- "And underlining" _ ";
- reduced limit of the size of the transmitted packet to 500 KB.

The following is an example of JSON (sgdi_rest_request_schema_v_0_6.json):

```
[
{
  "id": "1",
  "dev": "3520-080-9143_6268",
  "lat": 52.17237393,
  "lon": 21.1941930753,
  "tsp": 1539186656000000,
  "alt": 108.96,
  "acc": 24.0,
  "brg": 1.0,
  "spd": 1.0
},
{
  "id": "2",
  "dev": "3520-080-9143_6267",
  "lat": 51.17237393,
  "lon": 20.1941930753,
  "tsp": 1539186676000000,
  "alt": 8.96,
  "acc": 2.0,
  "brg": 360.0,
  "spd": 10.0
}
]
```

]

The data is transferred in the form of a JSON table, in which individual elements are JSON objects containing individual waypoints. The description of individual fields, the rules for validation and information about field maturity in Schema _v_0_6 is presented in Table 2.

Table 2. Schema_v_0_6

Name	Description	Validation rule	Required
id	The unique identifier of the record in the source system, the variable used for verification purposes during the testing period.	"type": "string", "minLength": 1,"maxLength": 32, "examples": ["1", "1960472"]	No
dev	The unique identifier of the locator, the maximum length of 50 characters is allowed, small and uppercase Latin letters are allowed from the ranges (a-z) and (A-Z), digits (0-9), hyphen-minus (-), underscore (_), which are a subset of the ASCII characters (American Standard Code for Information Interchange).	"type": "string", "minLength": 1, "maxLength": 50, "pattern": "^ [a-zA-Z0-9\\-_]{1,50}\$", "examples": ["000000000000B1", "35A058060495422C7934"]	Yes
lat	Latitude taken from the GPS transmitter, WGS reference system 84, recommended minimum number of decimal places: 6, maximum number of decimal places allowed: 10. The limits of assumed extreme coordinates: 49.0 N and 54.835778 N.	"type": "number", "minimum": 48.0, "maximum": 55.835778, "multipleOf": 0.0000000001, "examples": [52.0375868826, 52.172644]	Yes
lon	Longitude taken from the GPS transmitter, WGS reference system 84, recommended minimum number of decimal places: 6, maximum number of decimal places allowed: 10. The limits of the assumed outer coordinates: 14.116667 E and 24.15 E.	type": "number", "minimum": 13.116667, "maximum": 25.15, "multipleOf": 0.0000000001, "examples": [21.1956136, 20.026094]	Yes
alt	Ellipsoidal height taken from the GPS transmitter, unit [m], maximum number of decimal places allowed: 2.	"type": ["number", "null"], "minimum": -1000.0, "maximum": 4000.0, "multipleOf": 0.01, "examples": [10.0, 200.02]	No

tsp	Timestamp includes date and time taken from the GPS transmitter, associated with the geographical position of the record, UTC time zone, SENT GEO timestamp is similar to Epoch / Unix Timestamp, but given to the microsecond (16 digits), this is the number of microseconds that have passed since 00:00:00 Coordinated Universal Time (UTC), Thursday, 1 January 1970, the minimum value indicates 2017.09.20 00:00:00 UTC, integer.	"type": "integer", "minimum": 1505865600000000, "examples": [1506086623000000, 1511273867317000]	Yes
spd	Movement speed taken from the GPS transmitter - unit [m/s], maximum number of decimal places allowed: 2, maximum allowed speed: 56.00 [m/s].	"type": "number", "minimum": 0.0, "maximum": 56.0, "multipleOf": 0.01, "examples": [3.21, 20.0]	Yes
acc	Location accuracy taken from the GPS transmitter - circle radius in meters, maximum number of decimal places allowed: 2.	"type": "number", "minimum": 0.0, "multipleOf": 0.01, "examples": [10.14, 30.0]	Yes
brg	Azimuth - unit [grade], maximum number of decimal places allowed: 2.	"type": "number", "minimum": 0.0, "maximum": 360.0, "multipleOf": 0.01, "examples": [40.14, 230.0]	Yes

9 Sgdi_rest_request_schema_v_0_61.json

From April 1, 2019, issue 0.61 of the JSON scheme (sgdi_rest_request_schema_v_0_61.json) applies.

The introduced changes concern the rejection of data whose coordinates are from outside of Poland. Rules presents Table 3.

Table 3 *Rules for rejecting data from outside Poland*

Rule code	Rule	Comments
B-W06	if lon < 14.116667	Data rejection when the longitude is less than 14.116667. Refers to the western border.
B-S06	if lat < 49.0	Data rejection when latitude is less than 49.0. It concerns the southern border.

B-E06	if lon>24.15	Rejection of data when the longitude is less than 24.15 applies to the eastern border
B-N06	if lat > 54.835778	Data rejection when the latitude is greater than 54.835778. Refers to the northern border.
L-SSW-CZ	If the geographical coordinates meet the condition: $54.9 - \text{lat} - 0.3 * \text{lon} > 0$	Data rejection in the south-west. It applies to the border with the Czech Republic.
L-ESE-UA	If the geographical coordinates meet the condition: $1.25 * \text{lon} + 20.375 - \text{lat} > 0$	Data rejection in the south-east. Applies to the border with Ukraine.
S-NE-RU	If the geographical coordinates meet the condition: $\text{lon} > 19 \text{ AND } \text{lat} > 54.5$	Rejection on data in the north-east. Applies to the border with the Russian Federation.

The data is transferred in the form of a JSON table, in which individual elements are JSON objects containing individual waypoints. The description of individual fields, validation rules and field information in Schema _v_0_61 presents Table 4.

Table 4. Schema_v_0_61

Name	Description	Validation rule	Required
id	The unique identifier of the record in the source system, the variable used for verification purposes during the testing period.	"type": "string", "minLength": 1, "maxLength": 32, "examples": ["1", "1960472"]	No
dev	The unique identifier of the locator, the maximum length of 50 characters is allowed, small and uppercase Latin letters are allowed from the ranges (a-z) and (A-Z), digits (0-9), hyphen-minus (-), underscore (_), which are a subset of the ASCII characters (American Standard Code for Information Interchange). The size of letters is not distinguished.	"type": "string", "minLength": 1, "maxLength": 50, "pattern": "^[a-zA-Z0-9\\-_]{1,50}\$", "examples": ["000000000000B1", "35A058060495422C7934"]	Yes
lat	Latitude taken from the GPS transmitter, WGS reference system 84, recommended minimum number of decimal places: 6,	"type": "number", "minimum": -90.0, "maximum": 90.0, "multipleOf": 0.0000000001, "examples": [52.0375868826, 52.172644]	Yes

	maximum number of decimal places allowed: 10.	Rules for rejecting data from outside Poland	
lon	Longitude taken from the GPS transmitter, WGS reference system 84, recommended minimum number of decimal places: 6, maximum number of decimal places allowed: 10.	type: "number", "minimum": -180.0, "maximum": 180.0, "multipleOf": 0.0000000001, "examples": [21.1956136, 20.026094] Rules for rejecting data from outside Poland	Yes
alt	Ellipsoidal height taken from the GPS transmitter, unit [m], maximum number of decimal places allowed: 2.	"type": ["number", "null"], "minimum": -1000.0, "maximum": 4000.0, "multipleOf": 0.01, "examples": [10.0, 200.02]	No
tsp	Timestamp includes date and time taken from the GPS transmitter, associated with the geographical position of the record, UTC time zone, SENT GEO timestamp is similar to Epoch / Unix Timestamp, but given to the microsecond (16 digits), this is the number of microseconds that have passed since 00:00:00 Coordinated Universal Time (UTC), Thursday, 1 January 1970, the minimum value indicates 2017.09.20 00:00:00 UTC, integer.	type: "integer", "minimum": 1505865600000000, "examples": [1506086623000000, 1511273867317000]	Yes
spd	Movement speed taken from the GPS transmitter - unit [m/s], maximum number of decimal places allowed: 2, maximum allowed speed: 56.00 [m/s].	"type": "number", "minimum": 0.0, "maximum": 56.0, "multipleOf": 0.01, "examples": [3.21, 20.0]	Yes
acc	Location accuracy taken from the GPS transmitter - circle radius in meters, maximum number of decimal places allowed: 2.	"type": "number", "minimum": 0.0, "multipleOf": 0.01, "examples": [10.14, 30.0]	Yes
brg	Azimuth - unit [grade], maximum number of decimal places allowed: 2.	"type": "number", "minimum": 0.0, "maximum": 360.0, "multipleOf": 0.01, "examples": [40.14, 230.0]	Yes

10 Method of data transfer

Data for the SENT GEO data interface is transmitted using the REST mechanism using HTTPS and the HTTP POST method. Transmitted data should be included in the JSON structure in accordance with the JSON scheme constituting an attachment to this document. Each data sample collected during a single measurement that contains data (geographical coordinates of the means of transport - latitude and longitude, vehicle speed, azimuth of the transport mode, satellite data transmission error - location accuracy, ellipsoidal height) collected at the same time (date and time of acquisition coordinates - a time stamp containing the date and time) is passed as a single JSON object. In order to limit the number of transferred data packets, data from one vehicle or from different vehicles stored within a JSON object is sent as elements of the JSON table that creates a single data packet. A single JSON table can contain from 1 (verbally one) to 500 (in words five hundred) JSON objects. The maximum allowable size of a single packet in bytes is 1 MB (say one Mega Byte).

11 Security of transmitted data

Currently, data transmission to the SENT GEO interface is carried out only with the use of certificates. The security set includes a dedicated URL interface + access restriction for indicated IP + SSL / TLS + authorizations using the SSL / TLS client certificate.

12 Validation of data – duties on the part of ZSL Operator

The operator is required to validate the data packet using the currently valid JSON scheme before proceeding with its transmission to the SENT GEO data interface. Validation should be carried out using software that supports validation based on schemas in accordance with the version of the JSON Schema specification given in the currently binding JSON Schema of the SENT GEO data interface. The currently valid JSON diagram of the SENT GEO data interface is in agreement with the Schema JSON Draft-06 specification (<http://json-schema.org/draft-06/schema#>)

13 Data validation – list of messages

Regarding data validation, the basic rule is that any unrecognized packet should be re-sent unless it conflicts with JSON Schema, then it should be corrected (if possible) and re-sent (packets) irreplaceable should be omitted). This should be done in accordance with the HTTP code provided (https://en.wikipedia.org/wiki/List_of_HTTP_status_codes).

Table 5 contains the most common messages in the data validation process.

Table 5. List of frequently appearing messages

Message	Rule/Warning	Operator action
HTTP 200 JSON: {"result": "OK"}	confirmation of correct validation of the sent JSON package	Not required.
HTTP 200 JSON: {"result": "OK"} with warning	Verification of transmitted data. The data has been accepted by the system. Example: "warning": { "tsp": 1505865600000001, "msg": "The timestamp value is from the past.", "code": "tsp-past", "dev": "A19-AZ37WW-0", "now": 1546728686549000, "action": "pass" }	Check the correctness of the transmitted data.
HTTP 200 JSON: {"result": "OK"} with warning	Verification of transmitted data. The data was rejected by the system. Example: { "msg": "Unknown device. Expecting technical identifier, found something similar to GPS device number.", "code": "dev-not-tech", "dev": "A19-AZ37WW-0", "action": "drop" },	Check the correctness of the transmitted data.
HTTP 200 JSON: {"result": "OK"} with warning	Verification of transmitted data. The data was rejected by the system. Example: { "msg": "Unknown device.", "code": "dev-unknown", "dev": "identyfikator1", "action": "drop" },	Check the correctness of the transmitted data.
HTTP 200 JSON: {"result": "OK"} with warning	Verification of transmitted data. The data has been accepted by the system. Example: { "tsp": 2505865600000002, "msg": "The timestamp value is from the future.", }	Check the correctness of the transmitted data.

	<pre>"code": "tsp-future", "dev": "1", "now": 1546728686549000, "action": "pass" }</pre>	
<p>HTTP 200</p> <p>JSON:</p> <pre>{"result": "OK"}</pre> <p>with warning</p>	<p>Verification of transmitted data. The data was rejected by the system.</p> <p>Example:</p> <pre>{ "msg": "The device is located outside of Poland.", "code": "not-pl", "reason": { "rule": "lon < 14.116667", "lon": 12.0, "id": "B-W06", "lat": 50.0 }, "dev": "1234567890", "action": "drop" }</pre>	<p>Check the correctness of the transmitted data.</p>
SSL/TLS errors	data was not provided	The operator must check what happened.
<p>400 Bad Request</p>	the provided data packet does not conform to the current JSON scheme or does not meet any other requirements	
	if the data package contains many geolocation packages then	the operator must divide it into smaller packages and try to provide a split package.
	if any smaller packet generated an incompatibility error,	the operator must separate it as a packet with one geolocation position.
	If the operator can correct the wrong location of the geolocation,	he should do so and send the corrected geolocation position.
	if the operator is not able to correct the wrong geolocation or did it very late,	then in this situation the geolocation location should be removed.
	incorrect geolocation location or lack of it. There are single incidents something is wrong - data is not provided.	The operator should check what's happening.

	incorrect geolocation location or lack of it. There are many erroneous geolocation positions or there is a lack,	so such transport may not be monitored at all. Such cases will be punished from 01/01/2019, the operator must check what is happening.
	if the package contains several measurement points,	divide it into as many packages as there are messages and resend each packet.
	if one of the individual packages is rejected,	it should be sent after correcting the error or omitting it.
401 Unauthorized –	data was not provided	The operator must check what happened.
500 Internal Server Error -		The attempt should be repeated. The SENT GEO team must be informed of this case.
501 Not Implemented –	wrong http method	The operator must go to the POST or PUT method and try again.
503 Service Unavailable —	service unavailable	The operator should repeat the attempt to provide data until effective. The SENT GEO team should be notified in this situation.

ATTENTION:

Result = OK informs that the data is correct in the syntactic sense (meet the schema).

Each of the warnings is an independent result of the business rule. The action field determines what effect the given rule has on the data indicated in the warning. Rules with the "drop" action have a higher priority than those with the "pass" action.

Drop rules occur in the case of:

- 1) unregistered devices,
- 2) data from outside Poland.

In the case of these rules, this can be interpreted as the lack of a legal basis for processing the data indicated in the warning.

The tsp-past rule indicates that a record has been obtained from the past for the device.

Action = pass indicates that this rule is informative and does not result in ignoring the data.

The dev-unknown rule indicates that the device is not registered. Action = drop, on the other hand, indicates that this rule is a filter, so all data sent in the package will not be processed further. Often

the tsp-past rule is returned despite the data being rejected on the basis of another rule to inform that there is more than one problem with the data. In this case, it should be acknowledged that the devices in question were not delivered correctly to SENT GEO, as they relate to an unregistered GPS device, so the indicated location device can not be used to monitor the SENT transport. If the device you want to use is to be used to monitor the carriage, you must register it in the SENT_470 form.

14 Information necessary to connect the ZSL to SENT GEO

Currently, in order to connect the ZSL to SENT GEO, an advanced method based on certificates is used:

- a. based on the PUESC test portal forms (<https://test.puesc.gov.pl/en/web/puesc/strona-glowna>).
- b. Is supported by both the SENT team and the SENT GEO team.
- c. The ZSL operator behaves in the same way as in the case of procedures on the production portal (<https://puesc.gov.pl/en/web/puesc/strona-glowna>) only that using the SENT test documents on the test portal.

There are some technical details summary, which you should provide to your ZSL service Operator:

- A. both production and test SENT GEO data interfaces accept geolocation data which are delivered by HTTPS based REST-JSON mechanism with HTTP POST method;
- B. delivered data must be provided with JSON data structures which are compatible with provided current JSON schema – SENT GEO data interface validate delivered data against mandatory JSON Schema and reject any incompatible data;
- C. JSON Schema lets to delivered data in data packages, every package can include up to 500 geolocation positions for various geolocation devices or for the same geolocation device;

In order to receive data from ZSL devices, the **di.sent.itl.waw.pl** server was dedicated to the production environment. However, for the test environment it is **di-test.sent.itl.waw.pl**.

15 Application of certificates

The ZSL operator connects to the portal <https://test.puesc.gov.pl>. I assume that he already has an account. If not then he must do it. The same rules apply to the portal <https://puesc.gov.pl>. Then the window shown in Fig. 1 appears.

The screenshot shows the main window of the PUESC test portal. At the top, there is a header with logos of the Ministry of Finance, the State Research Institute of Connectivity, and the State Treasury Administration. Below the header, a navigation bar contains links such as 'Main page', 'Formularze', 'My documents', 'Customs Systems', 'Taryfa Celna', 'EORI', 'Newsletter', and 'Help'. The main content area features a large banner with the text 'PLATFORMA USŁUG ELEKTRONICZNYCH SKARBOWO-CELNYCH' and a globe graphic. To the right of the banner, there are several sections: 'FORMULARZE' (Registration, Renewal, Declaration), 'e-PRZEWOZ' (Description of services, Materials, Training), 'FAQ' (Help, Registration, Declaration), and 'KIEROWCO' (Check status, Declaration, Declaration). Below these sections, there is a large globe graphic. The right sidebar contains 'NAJCZĘŚCIEJ UŻYWANE' (Most used) links like 'AKCU' and 'PZASREQ', and 'KOMUNIKATY' (Communications) with dates. The footer includes logos for the Ministry of Finance, State Research Institute of Connectivity, and the European Union.

Fig. 1. The main window of the PUESC test portal

In the Forms tab, select the SENT Forms (Formularze SENT) from the drop-down list.

Another screen appears, which is shown in Fig. 2.

Fig. 2. Screen with SENT forms

In the right part of the screen, click on the **SENT-GEO tab - SERVICES AND ZSL DEVICES - Instructions for registering ZSL services and GPS devices related to the ZSL services**. Another window containing forms for ZSL is opened (Fig. 3).

Fig. 3. Screen with forms for ZSL

The ZSL operator selects the SENT460 form (Fig. 4).

Fig. 4. Screen with the SENT460 form

The individual fields of the form must be completed correctly. In field 4, you must paste the CSR (Certificate Signing Request).

CSR is generated based on your private key. You can use openssl (www.openssl.org). If the user already has a private key (e.g. private.key) then the command in the Linux environment has the following structure:

- `Openssl req -new -key private.key -out certificate.csr`

If the user does not have a private key, it can be generated, for example:

- `openssl genrsa -des3 -out tech-private.key 4096`

(4096 bits gives better security than a 2048 key)

An example of a file containing a private key is shown in **Fig. 5**.

```
-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEA77EQo66h5dj4n0wrgLG8J9JTheXkIHnyHdCeoh/oXt+cSAua
SvEsSeMUYYdw4fc0WeHUE55qNSphHeumgNZnyDP9vM4b+ZDWhhHeToWvwyY5iNXB
lmKuux1XP0tCsHXgPJ0ezrcbMTi5pM0QU9Fc4KKOpqIV65pjJ4IinMRlD4G3cPBD
dOOZqSmX7tHp97q+PbVbWwvUg6eISxsgQ16SZTbAoilaG8HgIO+5i2RRdZOJf++7
KGFjwEl+UxDgsNaSp7Au/UGUCzH51iQIh9N3Kfj+cGgroGv5q66kUI27d5VTZjyf
kW4k8gvltwueKScsc9/Ord1r6YopGg5xwQr+TQIDAQABAOIBAQDePSF9cqtF9X4I
TVqk16cqkQQqSU5sokTQSidbkRQmK1S/JCrqQ5VZ6Ldz+1260DCYiia2glpdcy7a
zCz01ldhtHsWfVBI5HdTLou2iJO/8Ig2DGQOgC8chQbpQ8HQ1WqVIBaF+ha3W64d
VJLH7f4ctfxoGi8S5XH8Jtgq3JoLdeH9YqaNzQ2LKSx91/PxO6J7sLya82KKUBrp
M3AoumtEt0YRy57JkV7j1YeYUFLpWT7cR5rh2cZs5r1fQTGQjQorWBu/e4Po7PMn
Vbp/qDBqnfemf/dxDWydtXtJukplmLdUSK15jAXApr2ZSXZ56espTnuIxkvvuzZ
mny15mItAoGBAP34wh8DZwvUeKIn408osSQzHEtMnefIMB0u0yoj94RQZuv8VwAR
eoTeFIEPOqggdB7MSgkgZpNuyYxw+OrQI4mM19Wh9DyHwnWTxN07pDJEb6BCukQb
/+bdjLSytmDyVhGm1MQ1E017MdnqrQSRURvByNRxbDzzOP7w1L2bASTAoGBAPGb
HIDD1xcHgzKdOWNof2RDE+UbgaU86aI3dtGSsoTo6bmPkXxf6PJPu8pLwzhVOafZ
EXH4qJ9CioE4r6PelyA944KDwx8mlBsU7E6fEchJaR6xykW8u25Nr5P304szxCTI
987eJmQq+BGUUp7LgC/QlcpIR7yyP+h5CNNkAp2fAoGAECsaiCLrzacSvX1+6KXX
Jsowm5ADqBiYTSJegZ88jNQ3LyFbUNToNm13D8Rp4DVzikiGOk7jXkMs9JWNGphv
NATAA4xkR6KW0F4Trvc8+tXx+WDNIqk75jmZCnwmn25yKxlrwJf1A97YFuQ+zF
rHT8Edt6a4vTEebGJJm62uMCgYAO6NMFH9AmqugrFW0/1lmh4oD01JB7WT8sUjD/
Gw7zwXgLSCLfLAnXhGrT1SEIoRAGsUE0RuHK07c0sBU3xhPlzghogqtAKCKn530
WcF7KxhqMGUrgH1LXpFkv5EEGwIJD14hA3EQeSxdNnJDI216ufiukMb6f2fK2JT
aMnp4QKBgDxQkHSX8E7Fh1Uijf3C8IMZsZ7frzCbdIfNX6/PcVrcx3UKSVWmB9/v
auOMEH2moo/FRZXdcZPI0wzcGb4oz4few2Dp2savew5QEGq4v3DZDEhGK5X7Yc+M
skL3MCgqGqVN1+fV4uFHzGqPpMKMXZHUKlpLTVWNvswe0SBfZ5U5
-----END RSA PRIVATE KEY-----
```

Fig. 5. Example of a private key file

In turn, an example of a file containing CSR is shown in **Fig. 6**.

```
-----BEGIN CERTIFICATE REQUEST-----
MIIClzCCAB8CAQAwZExCZAJBgNVBAYTA1BMMRQwEgYDVQQIDAtNQVpPV01FQ0tJ
RTERMA8GA1UEBwwIV0FSU1pBV0ExDDAKBgNVBAoMA05JVDELMAkGA1UECwwCWjYx
FzAVBgNVBAMMDnd3dy5pdGwud2F3LnBsMSUwIwYJKoZIhvcNAQkBFhZlLmtsaW1h
c2FyYUBpdGwud2F3LnBsMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA
77EQo66h5dj4n0wrgLG8J9JTheXkIHnyHdCeoh/oXt+cSAuaSvEsSeMUYYdw4fC0
WeHUe55qNSphHeumgNZnyDP9vM4b+ZDWhhHeToWvwyY5iNXB1mKuuxIXP0tCsHXg
PJOezrcbMTi5pM0QU9Fc4KKOpqIV65pjJ4IinMR1D4G3cPBDd0OZqSmX7tHp97q+
PbVbWwvUg6eISxsgQ16SZTbAo1aG8HgIO+5i2RRdZOFj++7KGFjwE1+UxDgsNaS
p7Au/UGUCzH51iQIh9N3Kfj+cGgroGv5q66kUI27d5VTZjyfkW4k8gvltwueKScs
c9/Ord1r6YopGg5xwQr+TQIDAQABoAAwDQYJKoZIhvcNAQELBQADggEBADjODu1l
Wqp2GJ/8nam/bjnh2WNSczQ0FjQ6IiK/+rh1BfOREky0J9cz+hRsZt5m9D8UVWkC
u4a/iJicrMZHPhTbC9tKuAk2c29ErXKJeSxR/anRKg9EbD7AB4RFmEjsJo/yRauL
oHetctQxNPDBspkCmo2eRrKb2LdhCGFQRG4Wx/Gg6iuzd7zZKnOVKMUELpOP/vTz
Gu6QUdi2kpg/cr5A1rwq4d5uIEag1vi9G8YXNa/wkqOrNsuP660Wj8u9QgIWpWdV
ikYJShahRRHFxk3Qr//3P3lg0vgc4AuDcs/r4aO1ET7dzuIt0qZymoQKPuOwXpfgY
gxjEmtwLRv5BgM8=
-----END CERTIFICATE REQUEST-----
```

Fig. 6. An example of a file containing CRS

More details can be found at:

<https://uk.godaddy.com/help/apache-generate-csr-certificate-signing-request-5269>

In field 7 of the SENT 460 form Return communication channels, check the checkbox next to E-mail and provide the e-mail address to which we will receive SENTA461. Of course, the SENT460 can not contain errors to be accepted by the system and finally the **Save** button should be pressed. An example of a SENT461 document is shown in Fig. 7.

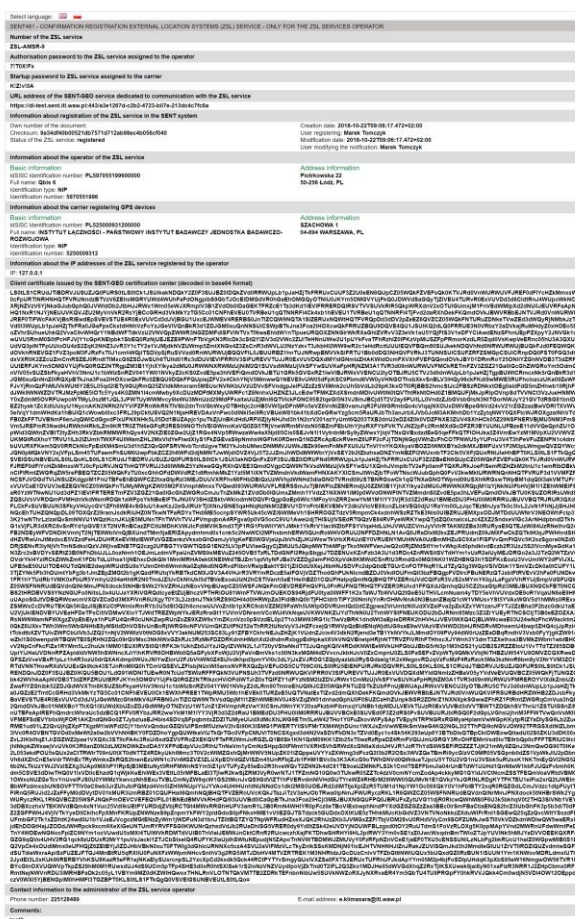


Fig. 7. An example SENT461

In SENT461, the ZSL Operator receives a Client Certificate encoded in the base64 format. It should be decoded. Do not add the BEGIN / END CERTIFICATE line to it, you only need to use a tool that can decode the text encoded in Base64, e.g.

- Notepad ++> Plug-ins> Mime Tools> Base64 Decode
- openssl base64 -d -in file_with_certificate_from_sent_461.txt -out certificate.pem
- Website <https://www.base64decode.org/>
- Certutil -decode file_with_certificate_from_sent_461.txt certificate.pem
(for Windows using the command line).

An example of a certificate in base64 is shown in Fig. 8.

```
LS0tLS1CRUdJTiBDRVJUSUZJQ0FUR50tLS0tCk1JSUVqekNDQW5jQ0FnRlhnQTBHQ1Nxr1NjYjNEUUVCC3dVQ
U1DQXhIakFjQmdOVk1JBTU1GVU5sY25ScFptbGokWVhSbE1FRjFkR2h2Y21sMGVUQWVGdZB4T0RBNUIUSXhNRE
V3TWpkYUZY2MHhPVEE1TVRJeE1ERXdnRmRtU1HRgpNUkF3RGdZRFZRUURFd2RvYjIxbExuQnNNU1l3RkFZRFZ
RUUtFdzFvYjIxbExuQnNjSE53TG1vdU1Rc3dDUV1EC1ZRUUdFd0pRVERFYk1Ca0dBMVVFQ0JNU2VtRmPhRz1r
Ym1sdmNHOXRiM0p6YTJsbE1SRXded11EV1FRSEV3aHoKZW1ON1pXTnBiakVjTUJvR0NTcUdTSWIZRFFFSkFSW
U5ZV1J0YVYc1QWFOXRaUzV3YkRDQ0FTSXdEUV1KS29aSQpodmNOQVFFQkRJRURnZ0VQQURDQ0FRb0NnZ0VCQU
1RMVp5Y1NnZ1hMRzRwSC9TWExvYwZTjVsa3NCcTFpcXorCmVucTBPMVUk0enRiRkYvZ1ZYWHPc1JwZEFnYWF
ieGNGZudTnZnZlYkVPMGtEeThjN1cVdmpMcVQwSGFuZEt3QUwKV1B5bndGaDAwR2RJRWJaTVRNTG1jbeZ4aU9B
NzhNd1Z5R3VzTTNSNwP2Y0tvQ204bWVpK2NVOEpOTENpwTdwQgpaRT1vZnN1RWXnd2Z1Mj10QWFMVTZOT1FVS
1QyQj1hUKIwMmJQVHZwQX1ldwE5VHpfK2h2ZjIyQ290Sm9FMXh6CkE0WHI0REFEM0dms1VDMnZmZ31UMHBkbm
c0e1Jpa1U5TGRpR05ja1VGMOFTUJQM1o3amZrMHgVw1JKRzg3dWIKZWJWM11DMEFRbj1vcURLcS9LRW15d3p
jw9wbHE1Nw1QVzZOQnFRDNNaHBiOWNjczZVQ0F3RUF8YU55TUhBdwpDUV1EV1IwVE1BSXdbREFkQmdOVkhR
NEVGZ1FVNGFqFRmekVtWmt1ZzJicRxejVSS1NrOWNVd0RnWURWUjBQckFRSC9CQVFEQWdPSU1CTUdBMVVKs
1FRtU1Bb0dDQ3NHQVFRk1J3TUNNQjhHQTfVZE13UV1NQmFBRk11bDlaQUQKbk81NERiOTQzd1dJNDUrc1Z3ck
NNQTBHQ1Nxr1NjYjNEUUVCC3dVQW5jQ0FnRlhnQTBHQ1Nxr1NjYjNEUUVCC3dVQW5jQ0FnRlhnQTBHQ1Nxr1NjYj
xUmXzRHNSU5wNH3WmkhvcmpPQUFddHcyan1NeU1obU1kOFJ1bm1hUUNSvUk4CnBXcXdhL1J0Q1J1dEdEdE0pH
bEjZdnR5bzVjZ3A2Tm9tVF85TE55WVhLMUJUWmo3RWZXRlg3aH10SGRWNHBAZC8KMTk0V2hucnR3S1Ubw1NV
Hkv13VubHhwbU9ieG9SmmRyZXkyOT1nYVROeThNbnVYNGNuNm03dmVsUbmRTVjKwptRGn4VUE5MjNLCX1jMm
V1M1FroVPndk5FanVES3d0eGhYNzMyRwdseG8yYk5IwMvPQVNBWVBBEFqJW1JdFQZCktUeXRKMcT1amo1dF1
hS2tRNkRSNGZSVUfUjErB2xTYj1TUTU3dkQ5RWc3ZUxabXhCQ3VdHhWZ2JuZVdTWfUKU1KL0h2UVhVWnQ0
aDc2RldOc01vdldYN1dCRWgzZ0thNjFDZTUybTRzY1h1YmpjMVBUtUE3eXRXKaUNEeGtoNQpSMW5wVVRkeF1oM
FdTcWNEUy8zS1lmVkJZe1Y0eHhZUWhvVH1VcndxNEt1M3p2bXN1V2k5bmZweXcvUEVpZTNRC1ZnUDRtUvpuYn
Byd1h1aUu5M2FzVnhDVk1VRZzZemhemNvVhd4YnZBeT1BZ1JGaeJ1S0g1TTE1Q0FrQUp3MwGkbk1CV3pXb3B
UY29EN1NxnuthVm84RVQyM29rZUpqMGY5Tk9EN1pOV2wrVzBSbk1aK0dYTKc0Z0FW50J1M3BibgphdWiyY1VvK
T1NmW5obU9aUudNwWtpSU0rR2IwdXpJdHdraEN1OSTwWWE4T2xvOFBPN2NTWHBSCUFpOFJJS3hDcndYbGwxV
1AyK3hhbHZsUnhdjhsVHZxc2VRPT0KLS0tLS1FTkQgQ0VSVE1GSUNBVEU0tLS0tLQo=
```

Fig. 8. Certificate encrypted in Base64

An example of a certificate decoded in PEM (Privacy-Enhanced Mail) format is shown in Fig. 9.

```

-----BEGIN CERTIFICATE-----
MIIDjCCBF6gAwIBAgICBEQwDQYJKoZIhvcNAQELBQAwge4xCzAJBgNVBAYTA1BM
MRQwEgYDVQQIDAttYXpvd211Y2tpZTE9MDsGA1UECgw0SW5zdH10dXQgYHEHWN6
bm/Fm2NpIC0gUGHFhHN0d293esSBJbnN0eXR1dCBcyWRhd2N6eTE8MDoGA1UECwwz
WmFrYXJhZCBaYWF3YW5zb3dhbnljYCBuZWNoZmlrIEluZm9yYWFjeWpueWNoIChh
LTYPMSkwJWYDVQQDDCBTRU5UIEdFTyBjVWwWlNMIjFRlc3QgTGv2ZWwgMSBDQTEh
MB8GCSqGSIb3DQEJARYSc2VudGdlb0BpdGwud2F3LnBsMBA4XDTE4MTAxODA3MDIw
NF0XDTE5MTAxODA3MDIwNFowZExCZAJBgNVBAYTA1BMMRQwEgYDVQQIDAttNqVp
V01FQ0tJRTERMA8GA1UEBwwIV0FSU1pBV0ExDDAKBgNVBAoMA05JVDELMAKGA1UE
CwwCWjYxYzFzAVBgNVBAMMDnd3dy5pdGwud2F3LnBsMSUwIwYJKoZIhvcNAQkBfH21
LmtsaW1hc2FyYUtpdGwud2F3LnBsMIIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA77EQo66h5dj4n0wrgLG8J9JTheXkIHnyHdCeoh/oXt+cSAuaSvEsSeMU
YYdw4fC0WeHUE55qNSphHeumgNZnyDP9vM4b+ZDWhhHeToWwvyY5iNXB1mKuux1X
P0tCsHXgPJ0ezrcbMTi5pMQU9F4cKKOpqIV65pjJ4IinMR1D4G3cPBDD0OZqSmX
7tHp97q+PbVbWwvUg6eISxsgQ16SZTbAoila8HgIO+5i2RRdZOFj++7KGFjwE1+
UxDgsNaSp7Au/UGUCzH51iQIh9N3Kfj+cGgroGv5q66kUI27d5VTZjyfkW4k8gv1
twueKScsc9/Ord1r6YopGg5xwQr+TQIDAQABo4IBdzCCAXMwCQYDVR0TBAlwADAd
BgNVHQ4EFgQUgzh3qIG1qOBurhVB9SH5iJ4nIUswDgYDVR0PAQH/BAQDAgXgMBMG
A1UdJQQMAoGCCSGAQUFBwMCMIIIBIAYDVR0jBIIBFzCCAROAFCwa4gqUtt+fYqFf
dRdBtFwmNS1poYH2pIHZMIHwMQswCQYDVQQGEWJQTEUMBIGA1UECAwLbWw6b3dp
ZWNraWUxETAPBgNVBACMCfDhcnN6YXdhMT0wOwYDVQQKDDRJBnN0eXR1dCdfGcSF
Y3pub8WbY2kgLSBQYcWEc3R3b3d5IEluc3R5dHV0IEJhZGF3Y3p5MTwvOgYDVQQL
DDBaYwFgmFkIFphYXdhbnNvd2FueWNoIFRlY2huaWwsgSW5mb3JtYWN5am55Y2gg
KFotNikxHTAbBgNVBAMMFNFNTlQgR0VPIElUTCBSb290IENBMRwwGgYJKoZIhvcN
AQkBfG16NkBPdGwud2F3LnBsggIAZANBgkqhkiG9w0BAQsFAAOCAEAEBN/Bj7HT
zSV+69+Q2uzWos+6tubKzJ8Eqv74s281WPhCGRYED2FID/3qLCN8kV+CpUoVaYoz
PWwr/oOednRDE/Aif2WnYb13UDxeWIFuSKx+ktY+NvqCaq9Jf1rmjZW56evZaRMs
xbYj0pju/cIg2PPj6UNH0hwdX6yvjv08vRS25JWY4UF0ekT5I6BMjfAEUbi75YXyK
yHkdhLiwiGRlHeQ4RVcodrPpn3+ojf07eidv3omHgQ7JmsGYCKu5ut4H7sGdOp28
tCuE0/IsrL7y4Suxo2uAR5RcW4COEPMTbKJh3XVvAYqKtH9dhGHu3ncR3F3TlqCO
NSxRWJ5JoNPxKTH4Pc8y/Ewalp+YX3wViJzeE8t2blb6aZocY+Hj2RA9Y13uG80Db
kRFcwp40Ht449Z2R/cZXkt23oc80uG1WQmzkz5BH6ZPuacQLdqEZ9ImTpcyUWE2A
rblxdNRB15QnzvFVBaXvBhzROgB812tArfMCiFvxlYwCTZvajNdyWbm5lQwWcXUv
jdZn3vwsPYru0/ImhN0ulP+YB1/XA09nfcTUax8pWmoJJvSgYlX8Y5fnYsEGD+Be
vO1E9wYubuoL4kX/56PxBa9ePXE/I4tYbF+9AGNsOHEslE1D5qN3yd13SgpHnR7
ueqBsmX+7yCq6KaFmiiJhKHko+Lq+6WY1hjCNUh7pp8cOZdAVFDNOiaOYdhCxU3
9u+FkpDYb01/sYjoVtKatwk+FEomoa/fQIcrml1Abvmk/J8XYf+SHmUR5h9pU0sv
hHmTUharftgtUjrkktgBWW1tNHqP+Fwk8tpsWh4M4r6cMJ1ShxJ+Xc+cfgTiJwcvE
otXX6ScZqlFmOgwUMLLnVJmN3zaycaaYjaHvIigiz8CVPomVaAtsaG70e9jKY7401
1ke47PRG3yGG456Rny1Wv38XBNpiWtTe+6NwLIEHSOPGIIpIuJnxsnio7bR1terY
i7m2nzPvbI9Qn/bFMLLnVjU51UR5RcFtb/p++pvlQuX5cf/rNANstBJT5mxdP7Du
m+TyEWxCMZWZI+h+0okJWmPqKBNg4tsTqHceiP7W2qZis0jZk162u/V6+ooQP891
AEtZaGkLc+Y/lg==
-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----
MIIKwjCCBqggAwIBAgICEAMwDQYJKoZIhvcNAQELBQAwfAxCzAJBgNVBAYTA1BM
MRQwEgYDVQQIDAttYXpvd211Y2tpZTERMA8GA1UEBwwIV2Fyc3phd2ExPTA7BgNV
BAoMNEluc3R5dHV0IMWBxIVjem5vxZtjaSAtIFBhXyRzdHdv3kgSW5zdH10dXQg
QmFkYXJhZCBaYWF3YW5zb3dhbnljYCBuZWNoZmlrIEluZm9yYWFjeWpueWNoIChh
LTYPMSkwJWYDVQQDDCBTRU5UIEdFTyBjVWwWlNMIjFRlc3QgTGv2ZWwgMSBDQTEh
MB8GCSqGSIb3DQEJARYSc2VudGdlb0BpdGwud2F3LnBsMBA4XDTE4MTAxODA3MDIw
NF0XDTE5MTAxODA3MDIwNFowZExCZAJBgNVBAYTA1BMMRQwEgYDVQQIDAttNqVp
V01FQ0tJRTERMA8GA1UEBwwIV0FSU1pBV0ExDDAKBgNVBAoMA05JVDELMAKGA1UE
CwwCWjYxYzFzAVBgNVBAMMDnd3dy5pdGwud2F3LnBsMSUwIwYJKoZIhvcNAQkBfH21
LmtsaW1hc2FyYUtpdGwud2F3LnBsMIIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA77EQo66h5dj4n0wrgLG8J9JTheXkIHnyHdCeoh/oXt+cSAuaSvEsSeMU
YYdw4fC0WeHUE55qNSphHeumgNZnyDP9vM4b+ZDWhhHeToWwvyY5iNXB1mKuux1X
P0tCsHXgPJ0ezrcbMTi5pMQU9F4cKKOpqIV65pjJ4IinMR1D4G3cPBDD0OZqSmX
7tHp97q+PbVbWwvUg6eISxsgQ16SZTbAoila8HgIO+5i2RRdZOFj++7KGFjwE1+
UxDgsNaSp7Au/UGUCzH51iQIh9N3Kfj+cGgroGv5q66kUI27d5VTZjyfkW4k8gv1
twueKScsc9/Ord1r6YopGg5xwQr+TQIDAQABo4IBdzCCAXMwCQYDVR0TBAlwADAd
BgNVHQ4EFgQUgzh3qIG1qOBurhVB9SH5iJ4nIUswDgYDVR0PAQH/BAQDAgXgMBMG
A1UdJQQMAoGCCSGAQUFBwMCMIIIBIAYDVR0jBIIBFzCCAROAFCwa4gqUtt+fYqFf
dRdBtFwmNS1poYH2pIHZMIHwMQswCQYDVQQGEWJQTEUMBIGA1UECAwLbWw6b3dp
ZWNraWUxETAPBgNVBACMCfDhcnN6YXdhMT0wOwYDVQQKDDRJBnN0eXR1dCdfGcSF
Y3pub8WbY2kgLSBQYcWEc3R3b3d5IEluc3R5dHV0IEJhZGF3Y3p5MTwvOgYDVQQL
DDBaYwFgmFkIFphYXdhbnNvd2FueWNoIFRlY2huaWwsgSW5mb3JtYWN5am55Y2gg
KFotNikxHTAbBgNVBAMMFNFNTlQgR0VPIElUTCBSb290IENBMRwwGgYJKoZIhvcN
AQkBfG16NkBPdGwud2F3LnBsggIAZANBgkqhkiG9w0BAQsFAAOCAEAEBN/Bj7HT
zSV+69+Q2uzWos+6tubKzJ8Eqv74s281WPhCGRYED2FID/3qLCN8kV+CpUoVaYoz
PWwr/oOednRDE/Aif2WnYb13UDxeWIFuSKx+ktY+NvqCaq9Jf1rmjZW56evZaRMs
xbYj0pju/cIg2PPj6UNH0hwdX6yvjv08vRS25JWY4UF0ekT5I6BMjfAEUbi75YXyK
yHkdhLiwiGRlHeQ4RVcodrPpn3+ojf07eidv3omHgQ7JmsGYCKu5ut4H7sGdOp28
tCuE0/IsrL7y4Suxo2uAR5RcW4COEPMTbKJh3XVvAYqKtH9dhGHu3ncR3F3TlqCO
NSxRWJ5JoNPxKTH4Pc8y/Ewalp+YX3wViJzeE8t2blb6aZocY+Hj2RA9Y13uG80Db
kRFcwp40Ht449Z2R/cZXkt23oc80uG1WQmzkz5BH6ZPuacQLdqEZ9ImTpcyUWE2A
rblxdNRB15QnzvFVBaXvBhzROgB812tArfMCiFvxlYwCTZvajNdyWbm5lQwWcXUv
jdZn3vwsPYru0/ImhN0ulP+YB1/XA09nfcTUax8pWmoJJvSgYlX8Y5fnYsEGD+Be
vO1E9wYubuoL4kX/56PxBa9ePXE/I4tYbF+9AGNsOHEslE1D5qN3yd13SgpHnR7
ueqBsmX+7yCq6KaFmiiJhKHko+Lq+6WY1hjCNUh7pp8cOZdAVFDNOiaOYdhCxU3
9u+FkpDYb01/sYjoVtKatwk+FEomoa/fQIcrml1Abvmk/J8XYf+SHmUR5h9pU0sv
hHmTUharftgtUjrkktgBWW1tNHqP+Fwk8tpsWh4M4r6cMJ1ShxJ+Xc+cfgTiJwcvE
otXX6ScZqlFmOgwUMLLnVJmN3zaycaaYjaHvIigiz8CVPomVaAtsaG70e9jKY7401
1ke47PRG3yGG456Rny1Wv38XBNpiWtTe+6NwLIEHSOPGIIpIuJnxsnio7bR1terY
i7m2nzPvbI9Qn/bFMLLnVjU51UR5RcFtb/p++pvlQuX5cf/rNANstBJT5mxdP7Du
m+TyEWxCMZWZI+h+0okJWmPqKBNg4tsTqHceiP7W2qZis0jZk162u/V6+ooQP891
AEtZaGkLc+Y/lg==
-----END CERTIFICATE-----

```

Fig. 9. An example of a decoded certificate

After decoding, a file containing a maximum of three PEM certificates is obtained:

- Customer certificate,
- CA Certificate (Authorization Center) level 1, which issued the client's certificate,
- CA certificate (Authorization Center) level 0, which issued a CA level 1 certificate.

Each certificate begins and ends with lines:

----- BEGIN CERTIFICATE -----
----- END CERTIFICATE -----

The above lines indicate the beginning and end of individual certificates.

The scope and manner of data usage, which are used to secure TLS communication, is different and depends on the system / application used by the entity. However, the typical requirements of SSL / TL tools / components include the use of the following elements during SSL authentication:

- client certificate;
- private key - which protects the possibility of using a client certificate only by the entity that is its administrator;
- the certification chain / certificate chain that authenticates the client's certificate as a certificate issued by the appropriate CA and contains:
 - CA certificate (Authorization Center) level 1, which issued the client's certificate,
 - CA certificate (Authorization Center) level 0, which issued a CA level 1 certificate.

In a Linux environment, the connection to the SENT GEO can be tested using the curl tool. The command sequence is shown below. Certificate.pem means the certificate obtained in SENT461, which was decoded from the base64 format to the PEM format. Whereas fd1.key means the private (decrypted) key used to generate CSR.

```
curl -X PUT --cert ./certificate.pem --key ./fd1.key -H 'Content-Type: application / json' -  
H 'cache-control: no-cache' -d '[{"id": "1960472", "dev": "ALBS8_74718", "years":  
52.17264488, "lon": 21.1956136, "alt": 140.0, "tsp": 1505893301000000, "spd": 0.0,  
"acc": 15.17, "brg": 0.0}, {"id": "1960473", "dev": "ALBS8_74718", "years":  
52.17264546, "lon": 21.195608, "alt": 138.0, "tsp": 1505896249000000, "spd": 10.0,  
"acc": 15.17, "brg": 0.0}]' https://di-test.sent.itl.waw.pl:443/10000000-0001-1001-0001-00000000000001
```

Note: The address <https://di-test.sent.itl.waw.pl:443/10000000-0001-1001-0001-00000000000001> should be replaced with the received address in SENT461, it is about the content of the **URL address of the SENT-GEO service dedicated to communication with the ZSL service.**

16 Tests

If the Carrier sending data to the SENT GEO system via the ZSL or the ZSL Operator want to test whether their data is correctly transmitted and stored in the SENT GEO TEST database, they can:

- Register as entities on the TEST PUESC portal (test.puesc.gov.pl), both the Carrier and the ZSL Operator,
- Register the ZSL service - only ZSL Operator.
- Register your test recorders:
 - Carrier,
 - ZSL Operator.
- Send location data to SENT GEO TEST,
 - from the Carrier's locators,
 - from the ZSL Operator locators.
- Verify:
 - in ZSL logs, does the response returned when transferring data to the JSON REST interface contain confirmation of correct data reception or information about the lack of non-compliance of transmitted data with the JSON scheme - only the ZSL Operator,
 - with the use of SENT 406 on the PUESC TEST, or the last saved in the SENT GEO TEST location of the locator corresponds to the transmitted data or not - Carrier for its locators, ZSL Operator for its locators,
 - with the use of the Mobile Driver's Application SENT GEO TEST, whether the last location stored in the SENT GEO TEST corresponds to the transmitted data or not - Carrier for its locators, ZSL Operator for its locators.

In the case of the production system SENT GEO, ZSL Operator may inspect logs, and the Carrier may use SENT 406 on the PUESC portal (puesc.gov.pl) and the Mobile Driver's Application SENT GEO in the production edition.

17 Contact

For help, please contact the HelpDesk via the following link:

<https://puesc.gov.pl/web/puesc/helpdesk-sc> or e-mail: helpdesk-eclo@mf.gov.pl

18 Good advice

In order to avoid the most common errors, it is recommended to:

- Validating the values of attributes included in a single data sample (a single JSON object) before it is added to the sample package (JSON table) so as to eliminate samples whose attributes do not meet the limits of the values given in the JSON scheme (eg samples from outside Poland) . Validation can be carried out in several ways:

- using the currently valid JSON schema, after encapsulating a single JSON object in the JSON table,
- using a slice of the currently valid JSON schema that applies to a single package,
- using programmatically supported validation of individual data in terms of their compliance with the rules defined in the JSON scheme.
- elimination of samples containing the location (latitude and longitude) of vehicles from non-defined Polish territory in the JSON scheme;
- specifying in vehicles the speed of vehicle movement expressed in meters per second ([m / s]), instead of incorrectly specifying the speed in km / h.

19 Interface for downloading the last location by ZSL operators

The interface for downloading the last location by ZSL operators has been made available only for production data.

In communication with the interface, use only TLSv1.2 secured HTTPS protocol.
The interface will be available at the address consistent with the pattern:

<https://di-status.sent.itl.waw.pl:<port>/<path>/<zsl>>

where:

<port>: is compatible with the port specified in the SENT-GEO service URL dedicated to communication with the ZSL service,

<path>: is compatible with the path specified in the URL of the SENT-GEO service dedicated to communication with the ZSL service,

<zsl>: is the ZSL service number.

For example, for the operator ZSL-XXXX-0 using the data interface:

<https://di.sent.itl.waw.pl:6666/abcdefgh-ijklmnop-rstuvwxyz>

the interface to verify the data transferred will be:

<https://di-status.sent.itl.waw.pl:6666/abcdefgh-ijklmnop-rstuvwxyz/ZSL-XXXX-0>

Access to the interface is possible only with the use of the current certificate (the last one obtained on the occasion of the creation or update of the ZSL operator data) and from the declared IP addresses.

The query may be performed no more frequently than once per hour + - 1 minute (tolerance introduced mainly due to time synchronization). In the case of more frequent queries, an

HTTP 429 error will be returned (example content further), and the time after which another query can be made will be extended by one minute. If the carrier changes the list of registered GPS devices related to the given ZSL service, it is possible to interrogate the interface after 5 minutes from the last poll.

In the case of a positive answer, JSON is returned in which:

- the *devices* object contains a list of objects, where the key is the technical identifier of the device, and the value of the object containing the sample time (tsp), length (lon), width (years), as well as the time of receiving the sample (recv); if there is no data for a given GPS device, these 4 variables are not returned,
- the *next* object contains the time in milliseconds in UTC, when the next query to the interface can take place.

Example of the correct answer:

```
{
  "next": 1552551621771000,
  "devices": {
    "dev-1": {},
    "dev-2": {},
    "dev-3": {
      "tsp": 1552351029000000,
      "recv": 1552351029000000,
      "lon": 21.123456,
      "lat": 52.123456
    },
    "dev-4": {
      "tsp": 1552551533000000,
      "recv": 1552551535000000,
      "lon": 20.654321,
      "lat": 50.654321
    }
  }
}
```

In the case of too frequent polling, an HTTP 429 error will be returned in JSON format, in which:

- the *result* object has the value "error",
- the *next* object contains the time in milliseconds in UTC, when the next query to the interface may take place,
- the *error* object contains an error message and an error code (429).

Example of HTTP 429 error:

```
{
  "result": "error",
```

```
"next": 1552552265475000,
"error": {
"msg": "To many request.",
"code": 429
}
}
```

Possible error codes together with descriptions that may occur can be found in Table 6.

Table 6. List of errors

Error code	Cause
HTTP 429 with JSON	Too frequent polling. Example above the table.
HTTP 429 without JSON	Too many connections from one IP address (more than 6 queries per minute).
HTTP 400 without JSON	An invalid HTTPS message has been sent.
HTTP 405 without JSON	A different HTTP method was used than GET.
HTTP 503 without JSON	Technical break.
HTTP 500 without JSON	An error occurred in the operation of the service.
HTTP 404 without JSON	Use the interface address that does not match the pattern, or use the address in which <path> or <zsl> has unrecognized syntax.
HTTP 401 without JSON	Authorization error caused by using a certificate other than that received in message 461.
HTTP 401 code = 401.3	The address uses an unregistered <path> value, please correct the address used.
HTTP 401 code = 401.4	The address uses the <path> value registered for a different port than the one specified in <port>, please correct the address used.
HTTP 401 code = 401.5	The address uses the <path> value registered for another ZSL operator than specified in <zsl>, the address used should be corrected. An example below the table.
HTTP 401 code = 401.6	The serial number of the certificate does not match the certificate registered for this address, use the last certificate received.
HTTP 401 code = 401.7	The fingerprint of the certificate does not match the certificate registered for this address, use the last certificate received.
HTTP 401 code = 401.8	Communication with the interface took place from an unregistered IP address, the list of addresses for the interface should be completed with the help of the SENT_463 message.

Example of error:

```
{
  "result": "error",
  "error": {
    "msg": "Access denied to Endpoint abcdefgh-ijklmnop-rstuvwxyz for ZSL with id ZSL-XXXX",
```

```
    "code": "401.5"  
  }  
}
```