

# Dokumentation der REST-Schnittstelle des Funk-Sensorsystem GesySense



**GesySense**

Gesytec GmbH  
Pascalstr. 6  
D 52076 Aachen

Tel. +(49) 24 08 / 9 44-0  
FAX +(49) 24 08 / 9 44-100  
e-mail: [info@gesytec.de](mailto:info@gesytec.de)  
[www.gesytec.de](http://www.gesytec.de)

Dokument / File	Version	Freigabe	Bearbeiter
REST-Schnittstelle Funk-Sensorsystem GesySense	01.10	09.11.17	Abdelaziz Boussema

## Vorbemerkungen

Dieses Dokument beschreibt die REST-Schnittstelle des Funk-Sensorsystems GesySense.

## Änderungsstand

Version	Abschnitt	Beschreibung der Änderung	Datum/Zeichen
01.00	Alle	Initiale Erstellung	12.10.2017/Ab
01.10	2	Schlüssel der CA-Zertifikate	09.11.2017/Ab
01.20	1	Systembild	28.11.2017/Ab

## Referenzdokumente

## Abkürzungen

CA	Certificate Authority

## Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b> .....	<b>5</b>
<b>2</b>	<b>Sicherheit</b> .....	<b>6</b>
2.1	Verschlüsselung.....	6
2.2	Authentifizierung.....	6
<b>3</b>	<b>Sensordaten</b> .....	<b>7</b>
3.1	URL .....	7
3.2	JSONObjekt.....	7
3.3	Antwort.....	8
3.4	Anzahl Objekt Pro POST .....	8
3.5	Fehlerverhalten .....	8
<b>4</b>	<b>Life-Sign</b> .....	<b>9</b>
4.1	URL .....	9
4.2	JSON.....	9
4.3	Antwort.....	9
4.4	Zeitverhalten .....	10
4.5	Fehlerverhalten .....	10
<b>5</b>	<b>Download</b> .....	<b>11</b>
5.1	Zeitverhalten .....	11
5.2	Fehlverhalten .....	11
<b>6</b>	<b>Begrenzung</b> .....	<b>12</b>

## 1 Einleitung

Dieses Dokument beschreibt die des Funk-Sensorsystems GesySense.

Der REST-basierte Dienst ermöglicht folgenden Datentransfer:

- Senden von Sensordaten an den Server
- Senden von Zustandsdaten des Receiver/LAN an den Server
- Herunterladen von Dateien z.B. Firmware-Update

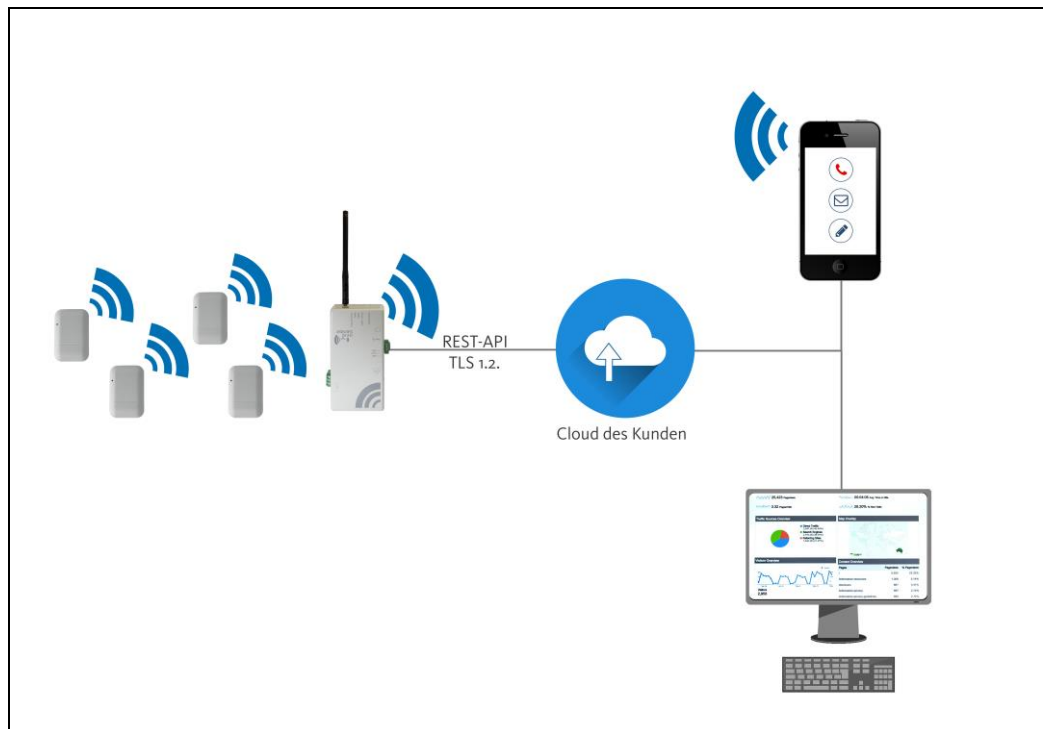


Abbildung 1 : Cloud Anbindung über RESTful Webservices

## 2 Sicherheit

### 2.1 Verschlüsselung

Der gesamte Datenverkehr zum und vom Receiver/LAN muss über Transport Layer Security (TLS) verschlüsselt werden. Hierzu wird das Protokoll Version TLS 1.2 verwendet.

Die REST-Schnittstelle unterstützt:

- Der Schlüssel der CA-Zertifikate:
  - **ECDSA-Schlüssel**
- Die Verschlüsselungs- und Signaturverfahren (Cipher-Suites):
  - **ECDHE-ECDSA-AES128-SHA256**

### 2.2 Authentifizierung

Nutzt man eine *Basic Access Authentication* in Kombination mit SSL/TLS via https, dann ist dieses Verfahren ausreichend geschützt.

Das Eintragen von Benutzername und Kennwort für den REST Client auf dem Receiver/LAN erfolgt über dessen Weboberfläche.

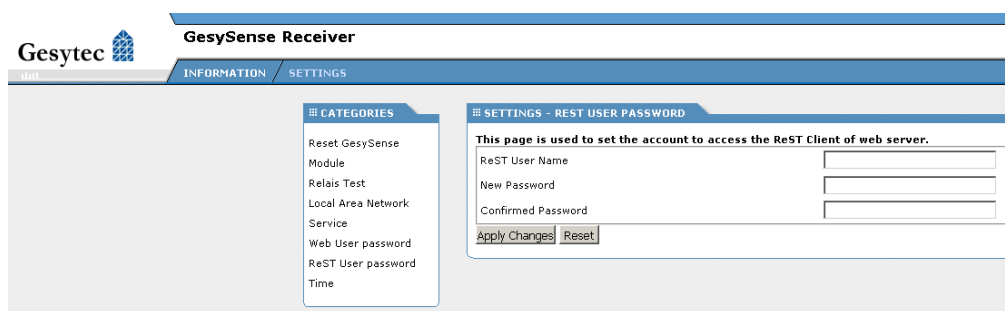


Abbildung 2 Weboberfläche

## 3 Sensordaten

### 3.1 URL

Die URL des REST Servers, an dem die Sensordaten geschickt werden, befindet sich in der Konfigurationsdatei *GRC.URL*. Diese Datei ist auf der SD Karte gespeichert.

### 3.2 JSONObjekt

Der Receiver/LAN sendet diese JSON-Objekte an den Server

```
{
  "receiver": "8000000012",
  "data": [
    {
      "sn": "0000015441",
      "value": "22.18",
      "type": "Temperature",
      "rssi": "78",
      "quality": "100",
      "battery": "100",
      "timestamp": "2017-09-26T14:43:04Z"
    },
    {
      "sn": "0000007989",
      "value": "-19.57",
      "type": "Temperature",
      "rssi": "66",
      "quality": "100",
      "battery": "99",
      "timestamp": "2017-09-26T14:43:06Z"
    }
  ]
}
```

Das JSON-Objekt enthält die Receiver/LAN-ID und die aktuellen Daten der jeweiligen Sensoren sowie die Seriennummer, den Messwert, den Typ des Messwertes, die Signalstärke, die Qualität des Signals, den Batterieladezustand und die aktuelle Zeit (**ISO 8601**) des Sensors.

Die Daten werden mit der HTTP-Methode *POST* gesendet.

### 3.3 Antwort

Auf einen erfolgreichen Request wird eine Antwort mit folgender Struktur erwartet:

```
{
  "response": {
    "status": " SUCCESS"
  }
}
```

### 3.4 Anzahl Objekt Pro POST

Die maximale Anzahl der gesendeten Objekte pro POST beträgt **32**.

### 3.5 Fehlerverhalten

Auf einen nicht erfolgreichen Request wird eine Antwort mit folgender Struktur erwartet:

```
{
  "response": {
    "status": " ERROR"
  }
}
```

Bei einer Unterbrechung der Verbindung zwischen dem Receiver und dem Server werden die Sensordaten gepuffert und bei Wiederherstellung der Verbindung gesendet.

## 4 Life-Sign

### 4.1 URL

Die URL des REST Servers, an dem die Zustandsdaten geschickt werden, befindet sich in der Konfigurationsdatei *UPDATE.URL*. Diese Datei ist auf der SD Karte gespeichert.

### 4.2 JSON

Der Receiver/LAN sendet zyklisch dieses Objekt an den Server:

```
{
  "GesysenseID": "8.000000000",
  "FWversion": "0.134",
  "BLversion": "1.4",
  "Repeaters": "3",
  "Sensors": "21",
  "SD Card Storage Total": "3942645760",
  "SD Card Storage Free": "1968078848",
  "Bootcnt": "502",
  "timestamp": "2017-10-04T08:47:30Z"
}
```

Der Server erhält die ID, die Firmware-Version, die Bootloader-Version und weitere Diagnoseinformationen wie die Zahl der im System bekannten Repeater, Sensoren, Daten über die SD Karte, die Anzahl der Bootvorgänge und die aktuelle Zeit (**ISO 8601**) des Receivers/LANs.

Die Daten werden mit der HTTP-Methode *POST* gesendet.

### 4.3 Antwort

Über die REST-Schnittstelle kann man ein Bootloader-Update oder ein Firmware-Update ausführen.

Zuerst vergleicht der Server die Bootloader-Version des Clients mit der aktuellen Version und gibt gegebenenfalls folgendes *fwupdate* Objekt in JSON Notation zurück, wenn die Version nicht aktuell ist:

```
{
  "fwupdate": {
    "URL": "https://rest1.gesysense.de/update/ttr4_bl_v1_04.bin"
  },
  "response": {
    "status": "SUCCESS"
  }
}
```



Ist die Bootloader-Version aktuell, wird nun auf das neueste FirmwareUpdate geprüft.

Der Server kann in der Antwort nun folgendes *fwupdate* Objekt in JSON Notation zurückgeben:

```
{
  "fwupdate": {
    "URL": "https://rest1.gesysense.de/update/TTR4_v0_132.bin"
  },
  "response": {
    "status": "SUCCESS"
  }
}
```

Der Server sendet die URL der Datei für das FW Update. Wichtig für den Receiver/LAN ist, dass der Dateiname mit *TTR4\_v* beginnt.

### 4.4 Zeitverhalten

Der Receiver sendet die Zustandsdaten alle **15 Minuten**.

### 4.5 Fehlerverhalten

Auf einen nicht erfolgreichen Request wird eine Antwort mit folgender Struktur erwartet:

```
{
  "response": {
    "status": " ERROR"
  }
}
```

### 5 Download

Das Herunterladen von Dateien (z.B. Firmware Update) auf den Receiver folgt mit dem HTTP Methode *GET*.

#### 5.1 Zeitverhalten

Der Receiver/LAN wertet die Antwort des HTTP Servers aus. Ist die Firmware-Update-URL enthalten, versucht der Receiver/LAN **unmittelbar** die angegebene Datei zu laden und führt bei erfolgreichem Laden auch **unmittelbar** das Update aus.

Soll das Update in einer bestimmten Zeit ausgeführt werden, legen Sie die Update-URL erst in dieser Zeit auf dem Server.

#### 5.2 Fehlverhalten

Nach **drei** erfolglosen Versuchen, die angegebene Datei herunterzuladen, wird das Verfahren gestoppt und im nächsten Zyklus (**15 Min**) durchgeführt.

## **6** **Begrenzung**

Die Zeichenlänge der URL muss kleiner als **128** Zeichen sein und die des JSONObjekts muss kleiner als **512** Zeichen sein.