

# TETRAcontrol

## TETRA-Funkgeräte Steuerung



## HTTP API

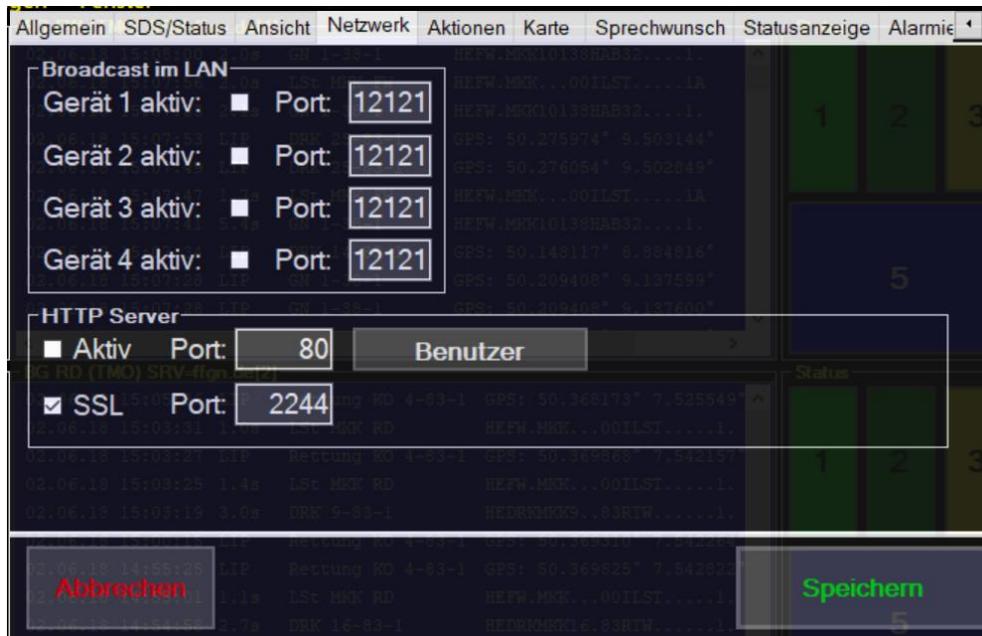
Status 3 IT GmbH

06/2020

Version 4.2

## 1.1 Web-Server

Der Web-Server wird über das *Optionen -> Netzwerk* Menü aktiviert.



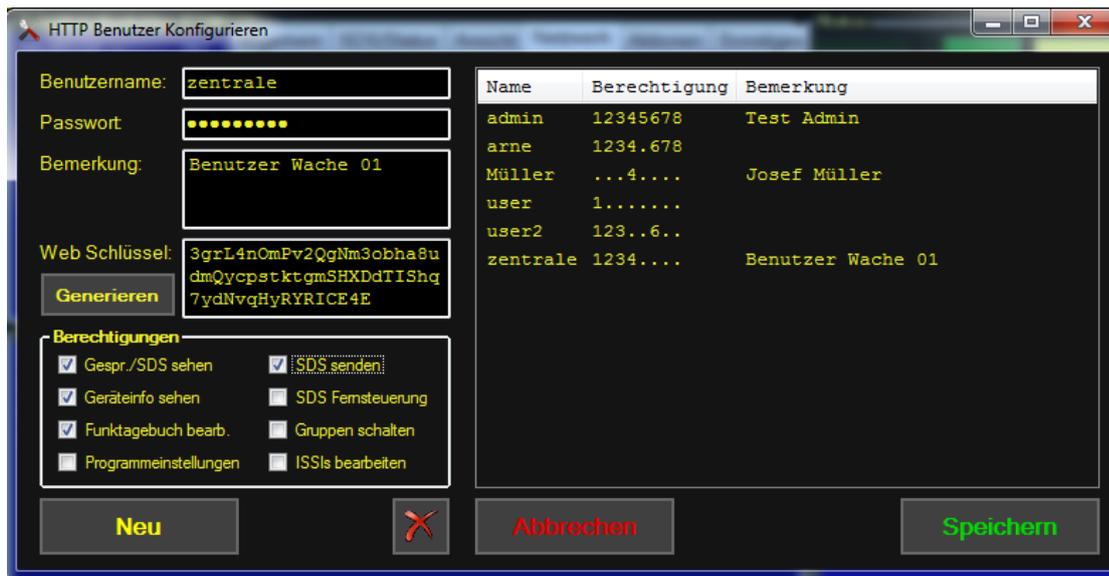
Nach der Aktivierung sollte TETRAcontrol einmal neu gestartet werden. Zusätzlich zum HTTP-Server kann auch ein HTTPS (SSL) Server gestartet werden, der die Daten verschlüsselt überträgt. Zum aktivieren ist der Haken bei SSL zu setzen und ein gültiger Port (Standard: 443) einzutragen. In TETRAcontrol wird ein selbst signiertes Zertifikat mitgeliefert. Dieses kann in aktuellen Webbrowsern zu Warnmeldungen führen. Wenn ein eigenes Zertifikat verwendet werden soll, kann dieses über die „Config.txt“ Datei eingebunden werden. Hierzu sind folgende Zeilen in der `config.txt` einzutragen:

```
SSLServerport=2244
SSLServerCertFile=Dateiname.pfx
SSLServerCertPW=PASSWORT
```

Die drei Werte sind den eigenen Gegebenheiten anzupassen. Ist die Zertifikatsdatei nicht mit Passwort geschützt, dann ist die Zeile trotzdem anzugeben und nach dem „=“ abzuschließen. Das Zertifikat muss im PKCS12 Format (in der Regel mit Dateiendung .pfx oder .p12) vorliegen.

Im ‚Netzwerk‘ Menü können über die ‚Benutzer‘ Schaltfläche die zugelassenen Benutzer mit ihren Zugriffsrechten konfiguriert werden.





Ein neuer Benutzer wird über die Schaltfläche ‚Neu‘ angelegt. Erst wenn diese ausgewählt wurde, kann der Benutzername editiert werden.

Die folgenden Berechtigungen können vergeben werden:

Gespräche / SDS sehen	Eingehende Gespräche/SDS/Status/Positionen anzeigen
Geräteinfo sehen	Gerätedaten (Seriennr., OPTA, Software Version) anzeigen
Funktagebuch bearbeiten	Bemerkungen zu Gesprächen und SDS hinzufügen
Programmeinstellungen	TETRAcontrol Einstellungen (Optionen) ändern, Lautstärke
SDS senden	SDS, Status und LIP Anforderungen senden
SDS Fernsteuerung	Die Funktion ‚SDS Remote Control‘ nutzen
Gruppen schalten	Gesprächsgruppen (GSSI) und TMP/DMO der Geräte umschalten
ISSIs bearbeiten	Die Liste der ISSIs ändern

## 1.2 Authentifizierung

Die Authentifizierung gegenüber dem Web-Server kann auf mehrere Arten erfolgen:

Benutzername/Passwort	Im http GET oder POST Request
Web Schlüssel (Userkey)	Im http GET oder POST Request, als http Cookie
Basic Authentication	Standard http Basic Authentication

Benutzername und Passwort werden als Parameter ‚user=‘ und ‚pw=‘ übergeben. Der Web Schlüssel als ‚userkey=‘

Die Authentifizierungs-Optionen werden in folgender Reihenfolge abgefragt:

- Benutzername / Passwort
- Web Schlüssel im Request
- Web Schlüssel im http Cookie



- Basic Authentication

Wenn ein Punkt fehlschlägt (d.h. Benutzer existiert nicht, Passwort falsch oder Keine Zugriffsrechte) dann wird die nächste Option versucht.

Bei einer erfolgreichen Authentifizierung wird ein http Cookie mit dem Web Schlüssel an den Client gesendet, so dass bei weiteren Aufrufen keine Authentifizierungs-Parameter in der URL übergeben werden müssen.

Die Datei ,login.htm[l]' und alle Dateien mit Endungen ,.js' und ,.css' können ohne Authentifizierung aufgerufen werden um eine Login-Seite zu ermöglichen.

**Beispiele:**

`http://192.168.99.112/API/statusgps.json?user=franz&pw=pa55w0rT`

`http://192.168.99.112/API/statusgps.json?userkey=akjDS24gjhGDg436fFFEEejkkO`

`http://franz:pa55w0rT@192.168.99.112/API/statusgps.json`



## 1.3 Bereitgestellte Adressen

### Websocket Protokoll:

/live.json                      Derzeit SDS/Gespräche/Status/GPS im JSON Format

Die Definitionen der JSON Datenpakete und Beispiele sind im Kapitel 2.1 und 2.2 zu finden.

### HTTP (statisch) Senden von SDS, Status, LIP:

/API/SDS  
/API/STATUS  
/API/LIP

#### Parameter:

Ziel	Ziel ISSI/GSSI
GerID	Gerätenummer (1-4)
Text	Text (für SDS)
StatusHEX	TETRA Status in Hexadezimal
StatusFMS	Status im FMS Format (1-8), wird in Hex umgewandelt
Typ	0=Einfache SDS, 1=Standard SDS, 195=CallOut, 138=verkettete SDS
Flash	0=Normale SDS, 1=Flash SDS. Stattdessen kann auch Typ=137 verwendet werden
Encr	0=Ohne Verschlüsselung, 1=Mit Verschlüsselung
Prio	[Optional] 0-15 Priorität, Default=0
COprio	1-15: CallOut Prio/Severity. Optional. Nur bei CallOut
CONum	1-250: Vorfall-Nummer. Optional. Nur bei CallOut
sub	Sub-Gruppen, mit Komma getrennt. Nur bei CallOut
remark	Bemerkung für internes Protokoll (nur bei SDS)
Best	1=Empfangsbestätigung anfordern

Der Parameter `sub` kann bei CallOut (`Typ=195`) angegeben werden. Alternativ können die Subgruppen auch dem Text vorangestellt werden. Bei der Angabe des `sub` Parameters erfolgt keine Längenprüfung, daher ist ggf der Text von der aufrufenden Applikation um die zusätzlichen Zeichen der Subgruppen zu kürzen. Falls die Zulässige SDS Textlänge überschritten wird, wird die SDS bzw die Alarmierung automatisch verkettet.

### HTTP (statisch) Listen im JSON Format:

/API/statusgps.json

#### Parameter:

<code>maxalterstatus</code>	in Sekunden
<code>maxalterpos</code>	in Sekunden

#### Beispiel:

/api/statusgps.json?maxalterstatus=86400&maxalterpos=3600



Wird ein Parameter nicht angegeben, werden die Standardwerte `maxalterstatus=2592000` (30 Tage), `maxalterpos=86400` (1 Tag) verwendet.

Es werden nur die ISSIs übertragen, bei denen Status bzw. Position nicht leer sind.

D.h. um alle ISSIs mit Status und/oder Position zu erhalten:

```
/api/statusgps.json?maxalterstatus=999999999&maxalterpos=999999999
```

Bitte beachten: Damit es keine undefinierten Timestamps gibt, werden diese per Default auf 1.1.2000 00:00 Uhr gesetzt. Somit sind alle mit `Jahr=2000` ungültig.

```
/API/issi.json
Parameter:
        filter      xxx
        issi        1234567
```

Der Parameter ‚Filter‘ sucht in ISSI, Name, OPTA und Bemerkung nach einem Substring-Match (Groß-/Kleinschreibung ist relevant!).

Wenn der Parameter ‚issi‘ angegeben wird, wird nur genau diese einzelne ISSI zurückgegeben. Wenn diese nicht existiert, eine leere Seite.

#### Beispiel:

```
http://192.168.1.1/api/issi.json?filter=DRK
```

#### HTTP (statisch) ISSI Status/GPS aktualisieren:

```
/API/ISSIUPD
Parameter:
        issi        ISSI die zu aktualisieren ist
        status      TETRA Status in Hexadezimal
        lat         Breite in Dezimalgrad
        lon         Länge in Dezimalgrad
        show        0 oder 1: Update anzeigen oder verstecken
```

Der Parameter „issi“ ist Pflicht. Mindestens status oder lat+lon müssen angegeben werden. Auch beide gleichzeitig möglich. Lat und Lon müssen in Dezimalgrad WGS84 angegeben werden, wobei der Dezimaltrenner ein Punkt („.“) sein muss.

Mit „show=0“ wird die ISSI aktualisiert, aber der Eintrag nicht in der Liste angezeigt.

Beispiel:

```
http://meintetracontrolserver.de/API/ISSIUPD?issi=1234567&status=8004&lat=50.1234&lon=9.4567&userkey=ABDHFkChjdadsf79878
```

#### HTTP (statisch) Gerätesteuerung:

```
/API/CONTROL
Parameter:
        cmd         Befehl. Derzeit implementiert:
                   tgs (Gruppe Schalten)
                   vol (Lautstärke)
        GerID       Gerätenummer (1-4)
        Param1     Parameter1
        Param2     Parameter2 (nicht benutzt)
```



Int1                      Numerischer Wert 1

Bei ,tgs' wird als Param1 die zu schaltende GSSI übergeben, entweder 7stellig für TMO oder 15stellig (26210...) für DMO.

Bei ,vol' als Int1 die Lautstärke (Maximum bei Motorola Geräten ist 14, bei Sepura 9)

Beispiel:

<http://meintetracontrolserver.de/API/CONTROL?cmd=tgs&GerID=1&Param1=2493302&userkey=ABDHFkChjdadsf79878>

#### HTTP (statisch) Geräteinformation (JSON Format):

/API/RADIO.JSON

Parameter:

GerID                      Gerätenummer (1-4)

Wenn ,GerID' nicht angegeben wird, wird als Wert 1 verwendet.

#### HTTP (statisch) ISSI.CSV Liste:

/API/ISSP.zip

Überträgt die ISSI.csv Liste des Servers im ZIP Format.

#### Allgemeine Parameter:

user	Benutzername
pw	Passwort
userkey	Web Schlüssel als Alternative zu Benutzername / Passwort
MaxAlter	Höchstalter (in Sek.) der beim ersten Aufruf übertragenen Daten
MaxAnz	Anzahl der beim ersten Aufruf übertragenen Daten

Weitere Seiten sind in Arbeit. Z.B. Statische (RESTful) Seiten für Status-Übersicht, Gespräche, SDS im JSON Format

Die von TETRAcontrol bereitgestellte Seite sieht derzeit folgendermaßen aus:



## Übersicht

Typ	Zeit	Gerät	Ziel	Dauer	Teilnehmer	OPTA/Meldung/Text
Status	09:03:33	FRT 1	3331001	Stat	9992002	Status 4 - Einsatzort an
Einzelruf	09:03:26	HRT 12	Einzelruf	von	9992003	Eingehender Einzelruf
Gespräch	09:02:56	MRT 2	3331002	00:11	9991001	TTDRKKRS12#83RTW.....11
Status	09:02:46	FRT 1	3331001	Stat	9992002	Status 3 - Einsatz übernommen
Gespräch	09:02:26	FRT 1	3331001	00:16	9992002	DIFW.KRS12146HLF20/16.1.
SDS	09:02:16	FRT 1	3331001	SDS	9992001	*Zimmerbrand*Florianstadt, Hauptstraße 49
Gespräch	09:01:57	MRT 2	3331001	00:11	9992002	DIFW.KRS12146HLF20/16.1.

SDS

Status

Gespräche

Zum Simulieren von Status + GPS Meldungen um eigene Entwicklungen zu testen, kann man mit der Tastenkombination SHIFT+STRG+E eine zufällige Generierung von Status und GPS in unregelmäßigen Zeitabständen auslösen.

## 1.4 Bereitstellung Routingdaten

Beschreibung folgt in Kürze.

## 1.5 TETRA Alarmgeber für divera247

TETRAcontrol kann Alarme von divera247 empfangen und diese als Callout aussenden.

Bei divera247 ist hier unter Verwaltung -> Schnittstellen -> Request-Service der „Request-Service“ mit der Internet-Adresse des TETRAcontrol SSL Servers zu konfigurieren. Als Format ist „POST application/json“ und als Inhalt „TETRAcontrol“ einzustellen. Die Authentifizierung am TETRAcontrol-Server kann entweder über den „userkey“ oder mit Benutzername/Passwort erfolgen, wobei die Userkey-Option zu empfehlen.

 Abbrechen  Speichern

SCHNITTSTELLE   ALARMSERVER   **REQUEST-SERVICE**   FIREBOARD   TETRACONTROL NBX

Der erzeugte Alarm-Datensatz kann über einen GET- oder POST-Aufruf automatisch an einen externen Dienst übergeben werden.

Alarmierung automatisch weiterleiten

URL/Ziel des Requests

Format

Inhalt

---

SCHNITTSTELLE   ALARMSERVER   **REQUEST-SERVICE**   FIREBOARD   TETRACONTROL NBX

Der erzeugte Alarm-Datensatz kann über einen GET- oder POST-Aufruf automatisch an einen externen Dienst übergeben werden.

Alarmierung automatisch weiterleiten

URL/Ziel des Requests

Format

Inhalt



Damit divera247 die Alarmer per HTTPS Request an den TETRAcontrol-Server senden kann, muss dieser aus dem Internet erreichbar sein. Dazu ist gegebenenfalls eine Port-Weiterleitung auf dem eigenen Internet-Router zu konfigurieren.



## 2.1 Anhang A – Objektdefinitionen

Die übertragenen Daten werden gemäß den untenstehenden Objektdefinitionen im JSON Format gesendet.

```
Class WSdev
  type As String = "radio"
  radioID As Integer = 0
  radioName As String
  active As Integer = 0
  vendor As String
  model As String
  serial As String
  swVersion As String
  ISSI As String
  OPTA As String
  TEI As String
  mode As Integer = -1 //0=TMO, 1=DMO
  GSSI As String
  GSSIname As String
  RSSI As Integer = 99
  cellID As Integer = 0
  cellName As String
  volume As Integer = -1
  ts As DateTime = Now //Zeitstempel
  enaLIP As Boolean = False //Location Information Protocol, ID 10
  enaSDSRC As Boolean = False //SDS Remote Control, usually 224
  enaHMD As Boolean = False //Home Mode Display, usually 220
  enaCallOut As Boolean = False //Alarmierung/Callout, usually 195
  colForeground As String = "#000000"
  colBackground As String = "#FFFFFF"
End Class

Class WSdevUpd
  type As String = "devupd"
  Index As Integer = -1
  CallActive As Boolean = False
  CallStatus As Integer = 5
  Mode As Integer = -1 //0 = TMO, 1=DMO
  GSSI As String
  CID As String
  RSSI As Integer = 99
  Volume As Integer = -1
  ScanStatusIST As Integer = -1
  ScanGroupColl As New List(Of TETRAscangroup)
End Class

Class WScmd
  type As String = "control"
  cmd As String
  Index As Integer = -1
  Param1 As String
  Param2 As String
  Int1 As Integer
End Class

Class WSpas
  type As String = "pos"
  srcSSI As String //Quell SSI des Status (Absender)
  srcName As String //Absender Name im Klartext
  Lat As Double //Breite; Nord > 0, Süd < 0
  Lon As Double //Länge; Ost > 0, West<0
  Alt As Double //Höhe
  FixQual As Integer = 0 //Qualität 0=Keine Pos, 1 => 100m, 2 <= 100m
  ts As DateTime = Now //Zeitstempel
End Class

Class WSsds
  type As String = "sds"
  destSSI As String //Ziel SSI der SDS (Empfänger, kann auch Gruppe sein, siehe //ServiceSelection!)
  destName As String //Empfänger Name im Klartext
  srcSSI As String //Quell SSI des Status (Absender)
  srcName As String //Absender Name im Klartext
  ProtocolID As Integer = 0 //2=SimpleText-SDS,9=Simple Flash SDS, 10=LIP, 130=Standard Text, 137=Flash SDS, 195=Callout
  text As String
  ts As DateTime = Now //Zeitstempel
  radioID As Integer = 1 //Nummer des an TETRAcontrol angeschlossenen Geräts (1-4)
  radioName As String //Name des Funkgeräts (z.B. 'F1')
  remark As String //Bemerkung. (Bei Callout steht hier Typ;Severity;Vorfall;Status / wobei Typ: -1=Alarmierung,
3=Rückmeldung; 5=Verfügbarkeit)
End Class
```



```

Class WSsendsds
  type As String = "sendsds"
  destSSI As String //Ziel SSI der SDS (Empfänger, kann auch Gruppe sein)
  ProtocolID As Integer = 0 //2=SimpleText-SDS,9=Simple Flash SDS, 10=LIP, 130=Standard Text, 137=Flash, 195=Callout
  text As String
  encrypt As Boolean = False
  receipt As Boolean = False
  prio As Integer = 0
  COprio As Integer = 3
  COvorfall As Integer = 0
  radioID As Integer = 1 //Nummer des an TETRAcontrol angeschlossenen Geräts (1-4)
End Class

Class WSstatus
  type As String = "status"
  status As String //Status (FMS) als Zahl bzw Buchstabe 0-9, A-F:
  //Achtung, nicht für alle TETRA Status gibt es ein Equivalent
  statusCode As String //Status als Hex Zahl aus der SDS
  statusText As String //Klartext des Status
  destSSI As String //Ziel SSI des Status (Empfänger)
  destName As String //Empfänger Name im Klartext
  srcSSI As String //Quell SSI des Status (Absender)
  srcName As String //Absender Name im Klartext
  ts As DateTime //Zeitstempel
  radioID As Integer = 1 //Nummer des an TETRAcontrol angeschlossenen Geräts (1-4)
  radioName As String //Name des Funkgeräts (z.B. 'F1')
  remark As String //Bemerkung
End Class

Class WScall
  type As String = "call"
  ID As String //Eindeutige ID um den Datensatz bei einer Aktualisierung (z.B. Gesprächsende) zu identifizieren
  destSSI As String //Ziel SSI des Gesprächs (Empfänger, kann auch Gruppe sein, siehe 'CallType')
  destName As String //Empfänger Name/Gruppe im Klartext
  srcSSI As String //Quell SSI des Status (Absender)
  srcName As String //Absender Name im Klartext
  srcOPTA As String //OPTA des Sendenden
  tsStart As DateTime = Now //TimeStamp Startzeit
  tsEnd As DateTime //TimeStamp Gesprächsende
  radioID As Integer = 1 //Nummer des an TETRAcontrol angeschlossenen Geräts (1-4)
  radioName As String //Name des Funkgeräts (z.B. 'F1')
  callType As Integer = 1 //0 = Point2Point, 1=GroupCall, 3= Broadcast(TMO), 4=Individual w/o presence check (DMO)
  priority As Int16 = 0
  remark As String //Bemerkung
End Class

Class WSissi
  type As String = "issi"
  ISSI As String //ISSI numerisch
  name As String //
  OPTA As String
  remark As String
  favorites As String //Favoritenlisten (1/2/3). Bsp: 13 = Favoritenliste 1 und 3
  lGSSI As String //Zuletzt aktiv in Gruppe (SSI)...
  lGSSIname As String //Letzte Aktive Gruppe (Name)
  lGSSITS As DateTime //Zeitstempel letzte Aktivität
  Lat As Double //Position WGS84 Nord/Süd
  Lon As Double //Position Ost/West (West < 0)
  postS As DateTime
  status As String //Letzter Fzg Status Hex Wert z.B. 8003
  statusTS As DateTime //Zeitstempel
  statusText As String //Status Klartext
End Class

```



## 2.2 Anhang B – Beispiele

Die JSON Pakete werden ohne Zeilenumbrüche übertragen.

### WSdev / Geräteinfo:

```
{
  "type": "radio",
  "radioID": 1,
  "radioName": "F4",
  "active": 1,
  "vendor": "MOTOROLA",
  "model": "MTM800FuG",
  "serial": "938TQG1234",
  "swVersion": "R27.220.8723",
  "ISSI": "4671234",
  "OPTA": "HEFW.XYZ10100.....1.",
  "TEI": "",
  "mode": 0,
  "GSSI": "2691121",
  "GSSIname": "BG FW",
  "RSSI": 30,
  "cellID": 12844,
  "cellName": "Frankfurter Str. 51",
  "volume": 12,
  "ts": "\/Date(1499155190641)\/",
  "enaLIP": true,
  "enaSDSRC": true,
  "enaHMD": false,
  "enaCallOut": true,
  "colForeground": "#FFFFFF",
  "colBackground": "#FF0000"
}
```

### WScall / TETRA Gespräch

```
{
  "type": "call",
  "ID": "2017070409144780441234402",
  "destSSI": "2691102",
  "destName": "BG RD",
  "srcSSI": "4123440",
  "srcName": "DRK 1-83-1",
  "srcOPTA": "HEDRKXYZ1..83RTW.....1.",
  "tsStart": "\/Date(1499152487839)\/",
  "tsEnd": "\/Date(1499152494057)\/",
  "radioID": 2,
  "radioName": "F5",
  "callType": -1,
  "priority": 0,
  "remark": ""
}
```

### WSStatus / TETRA Status:

```
{
  "type": "status",
  "status": "1",
  "statusCode": "8003",
  "statusText": "Status 1 - E-Bereit Funk",
  "destSSI": "2669900",
  "destName": "Daten",
  "srcSSI": "4671234",
  "srcName": "GG 2-42-1",
}
```



```
"ts": "\/Date(1499155242655)\/",  
"radioID": 1,  
"radioName": "Extern",  
"remark": ""  
}
```

**WSpos / GPS Position (LIP):**

```
{  
  "type": "pos",  
  "srcSSI": "4671234",  
  "srcName": "GG 2-42-1",  
  "Lat": 50.238576507568359,  
  "Lon": 8.05744457244873,  
  "Alt": 0,  
  "FixQual": 2,  
  "ts": "\/Date(1499155243155)\/"  
}
```



## 2.3 Kontakt

Wir freuen uns über Ihre Anfragen.

Kontaktadresse:

**Status 3 IT GmbH**

Dietrichsberg 33

63607 Wächtersbach

Fon: +49 (6051) 70076-00

Fax: +49 (6051) 70076-16

HRB 94696, RG Hanau

USt.ID. DE293587360

E-Mail: [tetracontrol@status3.it](mailto:tetracontrol@status3.it)

Web: <http://www.tetracontrol.de>

TETRAcontrol® ist eine eingetragene Marke der Status 3 IT GmbH

