

Contents

TRV Protocol — Reference Manual	1
1. Transport	1
2. General Frame Structure	1
3. Message Types	2
4. AP00 — Login	2
5. AP01 / AP10 — Location Update	2
6. CP01 — Heartbeat	4
7. Known SMS Commands	4
8. Practical Notes	6

TRV Protocol — Reference Manual

Scope: communication protocol used by cheap Chinese “mini” GPS/GSM trackers sold under names like **GF-21**, **GF-22**, and similar compact SOS/tracking pendants (sometimes labeled “S1903” in vendor docs). This protocol is informally called “**TRV**” because every message frame starts with that ASCII prefix. It is *not* an official/branded standard — it’s a manufacturer-shared, community-reverse-engineered protocol, mainly documented through the open-source [Traccar](#) GPS server project and its user forums.

This manual reflects what’s known from the Traccar [TrvProtocolDecoder](#) source and public forum traces. Coverage is not 100% complete — some fields (arm status, working mode) are only partially decoded even upstream, and some clones extend the format.

1. Transport

- **Protocol:** TCP (some clones/ports also accept the same text framing over UDP)
- **Common ports:** **5010**, **6100**, **8011** (varies by firmware/reseller)
- **Encoding:** plain ASCII text (not binary), one frame per message
- **Frame terminator:** **#** (hash character)
- **Keep-alive:** device sends a heartbeat (**CP01**) periodically (interval configurable, often every few minutes)

The device is configured (usually via SMS command) with the server domain/IP and port it should connect to — see §7.

2. General Frame Structure

TRV <TYPE> <PAYLOAD> #

- **TRV** — fixed 3-byte prefix identifying the protocol
- **<TYPE>** — a 4-character command code, always in the form **AP dd** or **CP dd** (letter, letter, 2 digits), e.g. **AP00**, **AP01**, **AP10**, **CP01**
 - **AP** = roughly “application”-class messages (login, location)
 - **CP** = “check/ping”-class messages (heartbeat)
- **<PAYLOAD>** — comma-separated fields, meaning depends on **<TYPE>**
- **#** — terminates every message in both directions

Server acknowledgement pattern

For every inbound message, the server is expected to reply with the **same type code but the first letter incremented by one** (a lightweight ACK), e.g.:

Device sends type	Server replies
AP00	TRVBP00#
AP01	TRVBP01#
CP01	TRVDP01#

(This is literally `char(type[0] + 1) + type[1:]`.) No payload is echoed back — just the incremented type and `#`.

3. Message Types

Type	Name	Direction	Purpose
AP00	Login	Device → Server	Sends device identifier (usually IMEI) to authenticate the session
AP01	Location update (interval)	Device → Server	Periodic GPS/LBS fix
AP10	Location update (alarm/event)	Device → Server	Same payload as AP01, sent on trigger (e.g. SOS, geofence, movement)
CP01	Heartbeat	Device → Server	Keep-alive with status fields, no GPS coordinates

4. AP00 — Login

TRVAP00<payload>#

The payload is the device's identifier (typically the 15-digit IMEI, sometimes with extra digits). The server uses this to associate the TCP session with a known device record. If the ID isn't recognized/provisioned, the server normally just drops the connection rather than sending further data.

5. AP01 / AP10 — Location Update

TRVAP<seq><DDMMYY><A/V><lat><N/S><lon><E/W><speed><HHMMSS><course><gsm><sats><batt><acc>xx

Real example (from a live device trace):

TRVYP14220217A5234.7885N00734.1840E000.0130919177.5610000506600002000
04,262,01,14635,52789,FritzBox7|DC-39-6F-7E-94-72|-89&...

Field-by-field breakdown

Field	Format	Example	Meaning
Date	DDMMYY	220217	22 Feb 2017
Validity	A / V	A	A = GPS fix valid, V = invalid (fall back to LBS/cell data)
Latitude	ddmm.mmmm	5234.7885N	Degrees + decimal minutes, NMEA-style. 52 deg 34.7885 min North
Longitude	dddmm.mmmm	00734.1840E	007 deg 34.1840 min East
Speed	ddd.d	000.0	km/h
Time	HHMMSS	130919 (example)	UTC time of fix
Course	6 chars, may include	177.561	Heading in degrees
GSM signal	ddd	000	Signal strength indicator (scale varies by firmware, often 0-31 or 0-100)
Satellites	ddd	050	Number of GPS satellites in view/used
Battery	ddd	066	Battery level, usually a percentage
ACC / ignition	d	0	0 = not applicable, 1 = ACC/ignition on, 2 = off (interpretation varies)
Arm status	dd	—	Arming/SOS state (not fully standardized across firmware)
Working mode	dd	—	Reporting mode config value
MCC	variable digits, comma-terminated	262	Mobile Country Code (e.g. 262 = Germany)
MNC	variable digits	01	Mobile Network Code
LAC	variable digits	14635	Location Area Code (cell tower LAC)
Cell ID	variable digits	52789	Cell tower ID
WiFi/extra	SSID\BSSID\RSSI&...	FritzBox7\DC-39-6F-7E	Optional nearby-WiFi list for WiFi-assisted positioning, &-delimited entries, each SSID MAC signal(dBm)

Coordinates convert to decimal degrees the standard NMEA way:
 decimal = degrees + (minutes / 60)

e.g. `5234.7885N` → `52 + 34.7885/60` ≈ `52.57981°N`

`AP10` uses the identical payload structure — the difference is purely semantic (it signals the report was event-triggered, e.g. SOS button, rather than the normal timed interval).

6. `CP01` — Heartbeat

TRVCP01,<gsm><gps><battery><acc><arm><mode>#

Field	Format	Meaning
GSM	<code>ddd</code>	Signal strength
GPS/satellites	<code>ddd</code>	Satellite count (no coordinates in this message)
Battery	<code>ddd</code>	Battery level
ACC	<code>d</code>	Ignition/ACC state
Arm status	<code>dd</code>	Arm/disarm state
Working mode	<code>dd</code>	Config mode

Heartbeats carry no position — they exist purely to keep the TCP session alive and report device health between location fixes.

7. Known SMS Commands

These are sent as plain **SMS text messages** directly to the tracker's SIM number — not over the `TRV` TCP link described in §2-§6. They're collected from published manuals for the GF-21/GF-22/GF-09 family and the related "S1903" firmware line. **Not every command works on every unit** — firmware forks between resellers, so treat this as a common reference, not a guarantee for a specific device. When in doubt, verify against the device's included manual or vendor support.

Example reported to work for pointing a GF-22 at a custom server:

`domain#46.101.24.212#5010#`

Server & connectivity

Command	Function
<code>domain#<ip>#<port>#</code>	Set the server address/port the device reports to
<code>apn#<parameter>#<account>#<password>#</code>	Set carrier APN (account/password optional)
<code>heartbeat#<seconds>#</code>	Set heartbeat interval (typically 10-120s)
<code>md#<seconds>#</code>	Set GPS reporting interval in seconds (typically 10-60s)
<code>time#<minutes>#</code>	Set GPS reporting interval in minutes (typically 1-120min)
<code>123#1</code> / <code>123#0</code>	Enable / disable real-time track upload (some firmware ships this off by default)

Device management

Command	Function
<code>imei#</code>	Query the device's IMEI number
<code>pwrst</code>	Reset device password to default
<code>1122</code>	Restore factory settings
<code>sysrst</code> / <code>SYSRST#</code>	Remote restart
<code>LED#on</code> / <code>LED#off</code>	Turn the status LED on/off (useful for making the device less visible)

Location & alerts

Command	Function
<code>999</code>	Request an SMS reply with a Google Maps link to current location
<code>000#<num1>#<num2>#<num3>#</code>	Bind up to three phone numbers for alerts/callback
<code>6666#on#</code> / <code>6666#off#</code>	Enable/disable vibration (shock) alarm
<code>666</code>	Send vibration alarm as an SMS to the bound number
<code>777</code>	Send vibration alarm as a phone call to bound numbers

Audio (device-dependent — many units include a mic for monitoring/SOS use)

Command	Function
<code>111</code>	Record 10 minutes of audio to TF card, then pause
<code>222</code>	Same as <code>111</code> , but auto-resumes another 10-minute recording if sound is detected
<code>333</code>	Voice-activated dial-back to bound numbers
<code>789</code>	Record 10 seconds of audio and upload to the app
<code>888</code>	Play back stored recordings (requires device speaker)
<code>444</code>	Delete all stored recordings
<code>monitor#<number>#</code>	Set a number that can silently call in to listen via the device mic
<code>monitor##</code>	Cancel the monitor/listen-in number
<code>555</code>	Disable all features

A related but distinct command set (`STATUS#`, `PARAM#`, `WHERE#`) shows up on GT06-protocol-family trackers rather than TRV-family ones — worth knowing if a device you're working with doesn't respond to any of the commands above, since it may be running the other protocol entirely.

8. Practical Notes

- **This isn't one official spec.** "TRV" is a de facto standard shared across many rebadged trackers (GF-21, GF-22, and others) from different factories, so field widths/behavior (especially "arm status" and "working mode") drift between firmware revisions. Treat the table above as the common core, and expect to verify against a live packet capture for any specific unit.
 - **No authentication beyond the ID string.** Login (`AP00`) just sends an identifier; there's no cryptographic handshake, so anyone who knows a device's ID and can reach the configured server/port can, in principle, inject data as that device. If you're standing up your own receiver, treat the channel as unauthenticated/untrusted at the transport level.
 - **Traccar** ships a working open-source decoder for this protocol (`TrvProtocolDecoder`), which is the most reliable reference implementation if you want to receive/parse this data programmatically rather than hand-roll a parser.
-

Sources

- Traccar open-source `TrvProtocolDecoder.java` (Apache 2.0 licensed)
- Traccar community forum threads: *"GF-22, TRV protocol, no gps location"*, *"Mini GF-21 Protocolo S1903"*, *"GF-22 (Port and usage)"*