

API DOCUMENTATION TRACKER.ID

Version 1.0 EN



API DOCUMENTATION, PARAMETER DATA and WEBHOOK TRACKER.ID

Web Login API

Web Login API allows to login into system via URL.(allows 3rd party web page login dialog integration).

Desktop :

Example

`https://tracker.id/login.php?username=USERNAME&password=PASSWORD&mobile=false`

Mobile :

Example

`https://tracker.id/login.php?username=USERNAME&password=PASSWORD&mobile=true`

Command Structure

HTTP variable "cmd" contains command and parameters. Command and parameters are split by comma. Command can have multiple parameters, every parameter is split by comma also.

Example

`https://tracker.id/api/api.php?api=user&key=ABCABCABCABC&cmd=OBJECT_GET_LOCATIONS,111111111`

OBJECT_GET_LOCATIONS – command
111111111- parameter

User API Commands

USER_GET_OBJECTS

Get object list of current user account.

Command structure

Command	USER_GET_OBJECTS

Example

`https://tracker.id/api/api.php?api=user&key=[API_KEY]&cmd=USER_GET_OBJECTS`

OBJECT_CMD_GPRS

Send ASCII or HEX command to object using GPRS gateway.

Command structure

Command	OBJECT_CMD_GPRS
Parameter #1	Object IMEI
Parameter #2	Command name
Parameter #3	Type (HEX or ASCII)
Parameter #4	Command, if command type is ASCII it is important to use quotes ("")

Example

```
https://tracker.id/api/api.php?api=user&key=[API_KEY]&cmd=OBJECT_CMD_GPRS,[IMEI],[Command name],HEX,COMMAND_TEXT
```

```
https://tracker.id/api/api.php?api=user&key=[API_KEY]&cmd=OBJECT_CMD_GPRS,[IMEI] , [Command name],ASCII,"COMMAND_TEXT"
```

OBJECT_GET_LOCATIONS

Get object(s) last know location data.

Command structure

Command	OBJECT_GET_LOCATIONS
Parameter #1	Object IMEI, for multiple IMEIs separate them via semicolon. Use "*" to get all available object locations.

Example

```
https://tracker.id/api/api.php?api=user&key=[API_KEY]&cmd= OBJECT_GET_LOCATIONS,[IMEI];[IMEI]
```

```
https://tracker.id/api/api.php?api=user&key=[API_KEY]&cmd= OBJECT_GET_LOCATIONS,*
```

OBJECT_GET_ROUTE

Get object trip history information including stops and events for set period.

Command structure

Command	OBJECT_GET_ROUTE
Parameter #1	Object IMEI
Parameter #2	Date and time from, format: YYYY-MM-DD HH:MM:SS
Parameter #3	Date and time to, format: YYYY-MM-DD HH:MM:SS
Parameter #4	Minimal stop duration in minutes (1, 2, 5, 10, 20, 30, 60, 120, 300)

Example

```
https://tracker.id/api/api.php?api=user&key=[API_KEY]&cmd=OBJECT_GET_ROUTE,[IMEI],2025-01-01 00:00:00,2025-02-01 00:00:00,[stop duration]
```

OBJECT_GET_EVENTS

Get object event messages for set period.

Command structure

Command	OBJECT_GET_EVENTS
Parameter #1	Object IMEI
Parameter #2	Date and time from, format: YYYY-MM-DD HH:MM:SS
Parameter #3	Date and time to, format: YYYY-MM-DD HH:MM:SS

Example

```
https://tracker.id/api/api.php?api=user&key=[API_KEY]&cmd=OBJECT_GET_EVENTS,[IMEI],2025-01-01 00:00:00,2025-02-01 00:00:00
```

OBJECT_GET_MAINTENANCE

Get Maintenance list of current user account.

Command structure	
Command	OBJECT_GET_MAINTENANCE
Parameter #1	Object IMEI, for multiple IMEIs separate them via semicolon. Use "*" to get all available object locations.

Example
<code>https://tracker.id/api/api.php?api=user&key=[API_KEY]&cmd=OBJECT_GET_MAINTENANCE,[IMEI]</code>

OBJECT_GET_EXPENSES

Get Expenses list of current user account.

Command structure	
Command	OBJECT_GET_EXPENSES
Parameter #1	Object IMEI, for multiple IMEIs separate them via semicolon. Use "*" to get all available object locations.
Parameter #2	Date and time from, format: YYYY-MM-DD HH:MM:SS
Parameter #3	Date and time to, format: YYYY-MM-DD HH:MM:SS

Example
<code>https://tracker.id/api/api.php?api=user&key=[API_KEY]&cmd=OBJECT_GET_EXPENSES,[IMEI],2025-01-01 00:00:00,2025-02-01 00:00:00</code>

Object Data Integration

Object data integration is useful for entering location data into tracker.id server.

Location structure	
imei	GPS object identifier
dt	Date and time, format: YYYY-MM-DD HH:MM:SS, time zone: 0 UTC
lat	Latitude
lng	Longitude
altitude	Altitude in meters
angle	Angle in degree
speed	Speed in km/h
loc_valid	1 is for valid GPS location, 0 is for invalid GPS location
params	Additional parameters for sensors, all should be split by " " character.
event	Possible events: sos, bracon, bracoff, dismount, disassem, door, mandown, shock, tow, pwrcut, gpscut, jamming, lowdc, lowbat, hacccl, hbrake, hcorn

Location

Example
<code>https://tracker.id/api/api_loc.php?imei=[IMEI]&dt=2016-01-01 00:00:00&lat=54.000000&lng=25.000000&altitude=100&angle=45&speed=60&loc_valid=1&params=batp=100 acc=1 </code>

Location + Event

Example

```
https://tracker.id//api/api_loc.php?imei=[IMEI]&dt=2016-01-01
00:00:00&lat=54.000000&lng=25.000000&altitude=100&angle=45&speed=60&loc_valid=1&params=batp=100
|acc=1|&event=sos
```

Parameter Data

Parameter Data Input Digital

CODE	DESCRIPTION
alaX	ala – Alarm, X refers to alarm number
acc	ACC status
bats	Battery status
diX	di – Digital Input, X refers to input number
doX	do – Digital Output, X refers to output number
pump	Fuel pump status
track	If device is in tracking mode or not
sleep	If device is in sleep mode or not
move	If device is moving or not
accel	Accelerometer status
dlog	Data logging status

Parameter Data Input Analog

CODE	DESCRIPTION
accv	ACC voltage
batl	Battery level
batp	Battery percentage
batv	Battery voltage
aiX	ai – Analog Input, X refers to input number
aoX	ao – Analog Output, X refers to output number
canX	can – CAN port data, X refers to CAN port number
tempX	t - temperature, X refers to temperature meter number
fuelX	fuel - Fuel level, X refers to fuel meter number
odo	Absolute odometer in km
odor	Relative odometer in km (between two records)
engh	Absolute engine hours
enghr	Relative engine hours (between two records)
gsmlev	GSM signal level, signal strength
gpslev	GPS signal level, number of satellites
hdop	Horizontal Dilution of Precision
pdop	Position [3D] Dilution of Precision
vdop	Vertical Dilution of Precision

Parameter Data Input String

CODE	DESCRIPTION
ibut	iButton for driver ID
rfid	RFID for driver ID
balance	SIM card balance
lac	GSM Location Area Code
cellid	GSM Cell ID
mcc	GSM Provider Country Code
mnc	GSM Provider Code
can_sysev	System Events - CAN
can_vemot	Vehicle Movement - CAN
can_clus	Clutch Pedal Status - CAN
can_bres	Brake Pedal Status - CAN
can_accs	Gas Pedal Status - CAN
can_crus	Cruise Control Status
can_ptos	PTO Status - CAN
can_enplcs	Machine PLCS Data
can_tyreloc	Tire Position Data - CAN
can_axlew	Wheel System Weight - CAN
can_axloc	Wheel System Position - CAN
can_haninf	Handling Information Data - CAN
can_dirind	Direction Indicator - CAN
can_flev	Fuel Level - CAN
can_fcr	Fuel Usage Assessment - CAN
can_feco	Economical Use of Fuel - CAN
can_pto	PTO Data - CAN
can_hrfc	HRFC Data - CAN
can_dist	Mileage Data - CAN
can_amt	Ambient Temperature - CAN
can_ect	Engine Temperature - CAN
can_rpm	Engine RPM Data - CAN
can_engh	Engine Working Hours - CAN
can_srdist	Mileage Before Maintenance - CAN
tco_dist	TCO Based on Mileage
tco_trip	TCO Based on Travel
tco_vespd	TCO Based on Speed
tco_rpm	TCO Based on RPM

Webhook

PHP GET

CODE	DESCRIPTION
<code>\$username = \$_GET['username'];</code>	User username
<code>\$name = \$_GET['name'];</code>	User's full name
<code>\$imei = \$_GET['imei'];</code>	IMEI of the device
<code>\$type = \$_GET['type'];</code>	Device type or data
<code>\$desc = \$_GET['desc'];</code>	Additional description
<code>\$zone_name = @\$_GET['zone_name'];</code>	Location zone name
<code>\$route_name = @\$_GET['route_name'];</code>	Travel route name
<code>\$lat = \$_GET['lat'];</code>	Latitude of the location
<code>\$lng = \$_GET['lng'];</code>	Longitude of location
<code>\$speed = \$_GET['speed'];</code>	Vehicle speed
<code>\$altitude = \$_GET['altitude'];</code>	Location altitude
<code>\$angle = \$_GET['angle'];</code>	Vehicle direction (angle)
<code>\$dt_server = \$_GET['dt_server'];</code>	Server time
<code>\$dt_tracker = \$_GET['dt_tracker'];</code>	Time on tracking device
<code>\$tr_model = \$_GET['tr_model'];</code>	Vehicle model
<code>\$vin = \$_GET['vin'];</code>	Vehicle identification number (VIN)
<code>\$plate_number = \$_GET['plate_number'];</code>	Vehicle plate number
<code>\$sim_number = @\$_GET['sim_number'];</code>	Driver's license card number
<code>\$driver_name = \$_GET['driver_name'];</code>	Driver's name
<code>\$trailer_name = \$_GET['trailer_name'];</code>	Trailer name (if applicable)
<code>\$odometer = \$_GET['odometer'];</code>	Mileage (odometer)
<code>\$eng_hours = \$_GET['eng_hours'];</code>	Engine running hours

PHP POST

CODE	DESCRIPTION
<code>\$username = \$_POST['username'];</code>	User username
<code>\$name = \$_POST['name'];</code>	User's full name
<code>\$imei = \$_POST['imei'];</code>	IMEI of the device
<code>\$type = \$_POST['type'];</code>	Device type or data
<code>\$desc = \$_POST['desc'];</code>	Additional description
<code>\$zone_name = @\$_POST['zone_name'];</code>	Location zone name
<code>\$route_name = @\$_POST['route_name'];</code>	Travel route name
<code>\$lat = \$_POST['lat'];</code>	Latitude of the location
<code>\$lng = \$_POST['lng'];</code>	Longitude of location
<code>\$speed = \$_POST['speed'];</code>	Vehicle speed
<code>\$altitude = \$_POST['altitude'];</code>	Location altitude
<code>\$angle = \$_POST['angle'];</code>	Vehicle direction (angle)
<code>\$dt_server = \$_POST['dt_server'];</code>	Server time
<code>\$dt_tracker = \$_POST['dt_tracker'];</code>	Time on tracking device
<code>\$tr_model = \$_POST['tr_model'];</code>	Vehicle model
<code>\$vin = \$_POST['vin'];</code>	Vehicle identification number (VIN)
<code>\$plate_number = \$_POST['plate_number'];</code>	Vehicle plate number
<code>\$sim_number = @\$_POST['sim_number'];</code>	Driver's license card number
<code>\$driver_name = \$_POST['driver_name'];</code>	Driver's name
<code>\$trailer_name = \$_POST['trailer_name'];</code>	Trailer name (if applicable)
<code>\$odometer = \$_POST['odometer'];</code>	Mileage (odometer)

\$eng_hours = \$_POST['eng_hours'];	Engine running hours
\$params = \$_POST['params'];	Additional parameters for data processing

This documentation will be continuously updated according to API developments and user needs.