TrackerServer RESTAPI
V2. 7

Prepared by	Dr.Noam Zeng	Audit	
Preparation Date	2013-5-13		

Version	Modification Date	Modify content	Remarks
V1.11	2018-04-16	Added temperature history interface	John Liu
V1.12	2018-05-13	Revised edition for TrackerServer2	John Liu
V1.13	2018-07-24	Add Transactions and History Trail	Noam Zeng
V2.2	2020-06-11	Added getHistoryTrailByUpdateIndex interface	John Liu
V2.3	2020-10-21	Update Documents	John Liu
V2.4	2021-12-16	Update Documents	Lan Zhou
V2.7	2023-01-03	Add Modify Relay Out Setting interface	Hui Guo

Table of Contents

Contents

1. Description

Tracker Server is a Java server developed by Anytrek to communicate with the tracker devices. The REST API defined here is for another server to communicate with the tracker devices via the Tracker Server. It has the following features:

1. Uniform Interface: it is Resource-Based, Manipulation of Resources Through Representations, Self-descriptive Messages.

2. Stateless: as REST is an acronym for REpresentational State Transfer, statelessness is key. The server does its processing, the appropriate state, or the piece(s) of state that matter, are communicated back to the client via headers, status and response body.

3. Cacheable: self-defined indexes to reduce client–server interactions, further improve scalability and performance.

4. Client-Server: the uniform interface separates clients from servers. Servers are not concerned with the user interface or user state, so that servers can be simpler and more scalable. Servers and clients may also be replaced and developed independently, as long as the interface is not altered.

5. Layered System: a client cannot ordinarily tell whether it is connected directly to the end server, or to an intermediary along the way. Intermediary servers may improve system scalability by enabling load-balancing and by providing shared caches. Layers may also enforce security policies.

2. Terms and Definitions

Acronym	Explanation
Device ID	Unique Device IDentifier, use IMEI or MEID
GPS	Global Positioning System
TCP	Transport Control Protocol
HTTP	HyperText Transfer Protocol
JSON	JavaScript Object Notation
LBS	Location Based Services
IMEI	International Mobile Equipment Identity for GSM system
MEID	Mobile Equipment Identifier for CDMA system
Int	Whole signed numbers only
String	Finite sequence of characters
Float	Numeric value including decimals
CRC	Cyclic Redundancy Check
uPulse	Data packet sent from device to server as status update and heartbeat
uRpLoc	Data packet sent from device to server as location and status update
Transaction	All data packet sent from device to server

Basic Data Type

U8	1Byte unsigned integer
S8	1Byte signed integer
U16	2Byte unsigned integer
S16	2Byte signed integer
U32	4Byte unsigned integer
S32	4Byte signed integer

3. Device -Oriented Query Interface

Resource representations of specific device/devices under a specific account. Useful for polling Info, Status and Settings of multiple devices in on request. Allow HTTP POST only.

3.1 Info List

Read Information of multiple devices.

1) URL

https://[*server_url*]/v2/api/info.json

2) Response Content Type

JSON

3) HTTP Method

GET/POST

4) HTTP Parameters

Parameter	Required	Type	Description
key	true	string	Access key v2, assigned by system, login Partner page to see
InvoiceId	false	string	

5) HTTP Body

A JSON array of device id, ex.: {"deviceId":["86392103007530","86392103007520",…]}

6) HTTP Response

A JSON array of device info objects with field definitions as below:

Field	Type	Length	Description
deviceId	String	14	Device ID
iccid	String	19	SIM iccid, reported from device
password	String	6	Tracker password
modelName	String		Product model name
hwVer	int		Hardware version
fwVer	int		Firmware version
blVer	int		Bootloader version
vehicleName	String		vehicleName
licensePlateNumber	String		Number plate
createTime	String		transaction recorded time in database ,yyyy-MM-ddTHH:mm:ssZ
ueVer	String		Modemversion

7) Example

Request:

Method: POST
URL: https://api.anytrek.com/v2/api/info.json
Parameters: key= 29e14d261efe493bb761423e66233ca1
Body: {"deviceId":["86727903860312","86727903568087"]}

Response:


```
[{
        "deviceId": 86727903860312,
         "hwVer": 110,
         "fwVer": 52,
         "ueVer":  null,
         "blVer": 3,
         "iccid": "89445017041889602 33",
         "password": "000000",
         "createTime": "2018-11-16T02:58:56+0000",
         "modelName": "VT1611",
         "vehicleName": "Trailer 311",
         "licensePlateNumber": null,
  }]
```


3.2 Status Lit

Read status of multiple devices.

1) URL

https://[*server_url*]/v2/api/status.json

2) Response Content Type

JSON

3) HTTP Method

GET/POST

4) HTTP Parameters

Parameter	Required	Type	Description
key	true	string	Access key v2, assigned by system, login Partner page to see

5) HTTP Body

A JSON array of device id, ex.: {“deviceId”:[“86392103007530”,” 86427503299472”,⋯]}

6) HTTP Response

A JSON array of device info objects with field definitions as below:

Field	Type	Length	Description
deviceId	String	14	Device ID
battery	float		Battery level in voltage
signals	int		0-100%
lat	float		latitude, six decimal places
lng	float		longitude, six decimal places
heading	int		in degree, 0-360, 0 means North
alt	int		altitude, in meters
speed	float		in km/h
locationTime	string		latest location update time in UTCeg,yyyy-MM-ddTHH:mm:ssZ
reportTime	string		latest communication time in UTCeg,yyyy-MM-ddTHH:mm:ssZ
charging	int		0 not charging, 1 charging
event	int		NULL, no data, 0 idle, 1 driving, 2 ACC off
totalMileage	int		Total mileage in meters; (Old Version)
odo	long		Total mileage in meters;
landmarkName	String		The name of the landmark where the vehicle is located
entryLandmarkTime	String		Entry landmark time in UTCeg,yyyy-MM-ddTHH:mm:ssZ
trailIndex	int		index of a trail point
vehicleName	String		vehicleName
iccid	String		SIM iccid, reported from device
extIn	int		External input
voltage	float		Residual Voltage
solar	int		Solar charging current
acc	int		Whether the transmitter is running, 0 is not running, 1 is running
temp	float		Temperature
hum	float		humidity
licensePlateNumber	String		number plate

7) Example

Request:

Method: POST
URL: https://api.anytrek.com/v2/api/status.json
Parameters: key= 29e14d261efe493bb761423e66233ca1
Body: {"deviceId":["86727903860312","86727903568087"]}

Response:


```
[
    {
        "deviceId": 86727903860312,
        "iccid": "89445017041889602 33",
        "locationTime": "2020-02-20T18:05:10+0000",
        "reportTime": "2020-02-20T18:05:10+0000",
        "alt": 0,
```

```
        "lat": 32.431255,
        "lng": -114.75918,
        "heading": 0,
        "speed": 0,
        "extIn": 0,
        "event": 0,
        "totalMileage": 67617887,
        "charging": 0,
        "battery": 100,
        "voltage": 4.11,
        "signals": 0,
        "solar": null,
        "temp": 0,
        "acc": 0,
        "hum": 0,
        "landmarkName": "Demo",
        "entryLandmarkTime": "2020-02-20T20:34:37+0000",
        "vehicleName": 14612565,
        "odo": “22218C”,
        "trailIndex": 19320,
        "licensePlateNumber": null,
    }
]
```


3.3 Single Device Status

Read status of specific device.

1) URL

https://[*server_url*]/v2/api/{deviceId}/status.json

2) Response Content Type

JSON

3) HTTP Method

GET

4) HTTP Parameters

Parameter	Required	Type	Description
key	true	string	Access key v2, assigned by system, login Partner page to see

5) HTTP Parameters

None

6) HTTP Response

A JSON array of history trail points, with field definitions as below:

Field	Type	Length	Description
deviceId	String	14	Device ID
battery	float		Battery level in voltage
signals	int		0-100%
lat	float		latitude, six decimal places
lng	float		longitude, six decimal places
heading	int		in degree, 0-360, 0 means North
alt	int		altitude, in meters
speed	float		in km/h
locationTime	string		latest location update time in UTC eg,yyyy-MM-ddTHH:mm:ssZ
reportTime	string		latest communication time in UTC eg,yyyy-MM-ddTHH:mm:ssZ
charging	int		0 not charging, 1 charging
event	int		NULL, no data, 0 idle, 1 driving, 2 ACC off
totalMileage	int		Total mileage in meters; (Old Version)
landmarkName	String		The name of the landmark where the vehicle is located
entryLandmarkTime	String		Entry landmark time in UTC eg,yyyy-MM-ddTHH:mm:ssZ
odo	long		Total mileage in meters;
trailIndex	Int		index of a trail point
vehicleName	String		vehicleName
iccid	String		SIM iccid, reported from device
extIn	int		External input
voltage	float		Residual Voltage
solar	int		Solar charging current
acc	int		Whether the transmitter is running, 0 is not running, 1 is running
temp	int		Temperature
hum	float		humidity
licensePlateNumber	String		number plate

7) Example

Request:

Method: GET
URL: https://api.anytrek.com/v2/api/867279038601231/status.json
Parameters: key= 29e14d261efe493bb761423e66233ca1
Body: none

Response:


```
{
    "deviceId": 86727903860312,
    "iccid": "89445017041889602 33",
    "locationTime": "2020-02-20T18:05:10+0000",
    "reportTime": "2020-02-20T18:05:10+0000",
    "alt": 0,
    "lat": 32.431255,
```

```
        "lng": -114.75918,
        "heading": 0,
        "speed": 0,
        "extIn": 0,
        "totalMileage": 67617887,
        "event": 0,
        "charging": 0,
        "battery": 100,
        "voltage": 4.11,
        "signals": 0,
        "solar": null,
        "temp": 0,
        "acc": 0,
        "hum": 0,
        "landmarkName": "Demo",
        "entryLandmarkTime": "2020-02-20T20:34:37+0000",
        "vehicleName": 14612565,
        "odo": “22218C”,
        "trailIndex": 19320,
        "licensePlateNumber": null,
}
```


3.4 History Trail

Read history trail points of specific devices.

1) URL

https://[server_url]/v2/api/{deviceId}/historyTrails.json

2) Response Content Type

JSON

3) HTTP Method

GET/POST

4) HTTP Parameters

Parameter	Required	Type	Description
key	true	string	Access key v2, assigned by system, login Partner page to see
startTime	true	string	ISO8601 format, yyyy-MM-ddTHH:mm:ssZ
endTime	true	string	ISO8601 format, yyyy-MM-ddTHH:mm:ssZ
deviceId	true	Long	Device ID

5) HTTP Body

None

6) HTTP Response

A JSON array of history trail points, with field definitions as below:

Field	Type	Length	Description
deviceId	String	14	Device ID
battery	float		Battery level in voltage
signals	int		0-100%
lat	float		latitude, six decimal places
lng	float		longitude, six decimal places
heading	int		in degree, 0-360, 0 means North
alt	int		altitude, in meters
speed	float		in km/h
reportTime	string		latest communication time in UTC eg,yyyy-MM-ddTHH:mm:ssZ
createTime	string		When the server received the data (UTC) eg,yyyy-MM-ddTHH:mm:ssZ
fwVer	int		firmware version. it may be null.
blVer	int		boot loader version, it may be null
charging	int		0 not charging, 1 charging
event	int		NULL, no data, 0 idle, 1 driving, 2 ACC off
totalMileage	int		Total mileage in meters;
updateIndex	int		index of a trail point
gpsState	Int		Location status: 0 not located, 1 2 3 located
hdp	int		Horizontal accuracy -1 indicates no record
accuracy	float		positional accuracy
country	String		address
state	String		About the address
city	String		city
streetAddress	String		Street Address
zip	String		Detailed Address
temp	int		Temperature
acc	int		Whether the transmitter is running, 0 is not running, 1 is running
voltage	float		Residual Voltage
solar	int		Solar charging current
current	int		charge current

7) Example

Request:

Method: GET

URL: https://api.anytrek.com/v2/api/86660604893314/historyTrails.json

Parameters: key= c0d5f837524a44b9a030736ec1b9eb63&startTime=2019-11-04T07:00:00-0000&endTime=2019-11-05T07:00:00-0000

Response:


```
[{
    "deviceId": 86660604893314,
    "gpsState": 1,
    "lat": 33.77822,
```

```
        "lng": -118.24782,
        "hdp": null,
        "heading": 0,
        "speed": 0,
        "accuracy": 0.0,
        "alt": 0,
        "country": null,
        "state": null,
        "city": null,
        "streetAddress": null,
        "zip": null,
        "totalMileage": 67617887,
        "temp": 0,
        "charging": 1,
        "acc": 0,
        "battery": 100,
        "voltage": null,
        "signals": 90,
        "solar": null,
        "event": 0,
        "current": 315,
        "fwVer": 48,
        "blVer": 4,
        "reportTime": "2019-11-04T07:57:19+0000",
        "createTime": "2019-11-04T07:57:20+0000",
       "updateIndex": 19320
}]
```


3.5 History Trail By UpdateIndex

Read history trail points of specific devices.

1) URL

https://[*server_url*]/v2/api/{deviceId}/historyTrails.json?_method=byUpdateIndex

2) Response Content Type

JSON

3) HTTP Method

GET/POST

4) HTTP Parameters

Parameter	Required	Type	Description
_method	true	string	byUpdateIndex
key	true	string	Access key v2, assigned by system, login Partner page to see
updateIndex	true	int	index of a trail point
count	false	int	Maximum number of transactions requested Default: 1000 Ranges: 1-10000

5) HTTP Body

None

6) HTTP Response

A JSON array of history trail points, with field definitions as below:

Field	Type	Length	Description
deviceId	String	14	Device ID
battery	float		Battery level in voltage
signals	int		0-100%
lat	float		latitude, six decimal places
lng	float		longitude, six decimal places
heading	int		in degree, 0-360, 0 means North
alt	int		altitude, in meters
speed	float		in km/h
reportTime	string		latest communication time in UTC eg,yyyy-MM-ddTHH:mm:ssZ
createTime	string		When the server received the data (UTC) eg,yyyy-MM-ddTHH:mm:ssZ
fwVer	int		firmware version. it may be null.
blVer	int		boot loader version, it may be null
charging	int		0 not charging, 1 charging
event	int		NULL, no data, 0 idle, 1 driving, 2 ACC off
totalMileage	int		Total mileage in meters;
updateIndex	int		index of a trail point
gpsState	Int		Location status: 0 not located, 1 2 3 located
hdp	int		Horizontal accuracy -1 indicates no record
accuracy	float		positional accuracy
country	String		address
state	String		About the address
city	String		city
streetAddress	String		Street Address
zip	String		Detailed Address
temp	int		Temperature
acc	int		Whether the transmitter is running, 0 is not running, 1 is running
voltage	float		Residual Voltage
solar	int		Solar charging current
current	int		charge current

7) Example

Request:

Method: POST

URL: https://api.anytrek.com/v2/api/86610402349293/historyTrails.json

Parameters:

key=c0d5f837524a44b9a030736ec1b9eb63&updateIndex=19320&count=100&_method=byU pdateIndex

Body:

Response:

```
[{
        "deviceId": 86610402349293,
         "gpsState": 1,
         "lat": 33.77822,
         "lng": -118.24782,
         "hdp": null,
         "heading": 0,
         "speed": 0,
         "accuracy": 0.0,
         "alt": 0,
         "country": null,
         "state": null,
         "city": null,
         "streetAddress": null,
         "zip": null,
         "totalMileage": 67617887,
         "temp": 0,
         "charging": 1,
         "acc": 0,
         "battery": 100,
         "voltage": null,
         "signals": 90,
         "solar": null,
         "event": 0,
         "current": 315,
         "fwVer": 48,
         "blVer": 4,
         "reportTime": "2019-11-04T07:57:19+0000",
         "createTime": "2019-11-04T07:57:20+0000",
         "updateIndex": 19320,

  }]
```


3.6 Trip Report

Read trip report data of specific devices.

1) URL

https://[server_url]/v2/api/{deviceId}/tripReport.json

2) Response Content Type

JSON

3) Response Content Type

GET

4) HTTP Parameters

Parameter	Required	Type	Description
key	true	string	Access key v2, assigned by system, login Partner page to see
startTime	true	string	ISO8601 format, yyyy-MM-ddTHH:mm:ssZ
endTime	true	string	ISO8601 format, yyyy-MM-ddTHH:mm:ssZ

5) HTTP Body

None

6) HTTP Response

A JSON array of device info objects with field definitions as below:

Parameter	Type	Description
totalDuration	int	unit: second
totalDrivingTime	int	unit: second
totalMileage	int	unit: meter
states	json	{du, state, mi} du: duration, unit: second state: State Abbreviation mi: mileage, unit: meter
days	array	Mileage statistics for each day

7) Example

Request:

Method: GET
URL https://api.anytrek.com/v2/api/86805004225121/tripReport.json
Parameters key=icpad2iipad2xx7ad2i7ad25tub2u&startTime=2022-01-02T14:00:00-0000&endTime=2022-01-22T17:00:00-0000

Response:


```
{
    "totalDuration": 92844000,
    "totalDrivingTime": 0,
    "days": [
        {
            "date": "2022-08-01",
            "states": []
        }
    ],
    "totalMileage": 0,
    "states": [
        {
            "du": 92844000,
            "state": "MS",
            "mi": 0
        }
    ]
}
```


3.7 Get Setting

Get the setting parameters of the device.

1) URL

https://[*server_url*]/v2/api/setting.json

2) Response Content Type

JSON

3) Response Content Type

GET

4) HTTP Parameters

Parameter	Required	Type	Description
key	true	string	Access key v2, assigned by system, login Partner page to see
deviceId	true	string	Device ID

5) HTTP Body

None

6) HTTP Response

A JSON array of device info objects with field definitions as below:

Parameter	Required	Type	Description
extIn	true	int	External input
updateFlag	true	int	Is there any update, 1024-yes, 0-no
relayOut	true	int	Relay control output, 0-close,1-open

7) Example

Request:

Method: GET
URL https://api.anytrek.com/v2/api/setting.json
Parameters key=453o333o333o34ysn5w&deviceId=86427503640822

Response:

```
[
    {
        "extIn": 0,
        "updateFlag": 1,
        "relayOut": 0,
    }
]
```

3.8 Modify Relay Out Setting

Modify the device relay output settings.

1) URL

https://[*server_url*]/v2/api/setting.json

2) Response Content Type

JSON

3) Response Content Type

POST

4) HTTP Parameters

Parameter	Required	Type	Description
key	true	string	Access key v2, assigned by system, login Partner page to see
deviceId	true	string	Device ID
relayOut	true	int	The control output of the 1508D relay, each bit represents a switch, 0 or 1

5) HTTP Body

None

6) HTTP Response

A JSON array of device info objects with field definitions as below:

Parameter	Required	Type	Description
extIn	true	int	External input
updateFlag	true	int	Is there any update, 1024-yes, 0-no
relayOut	true	int	The control output of the 1508D relay, each bit represents a switch, 0 or 1

7) Example

Request:

Method: POST
URL https://api.anytrek.com/v2/api/setting.json
Parameters key=453o333o333o34ysn5w&deviceId=86427503640822&relayOut=0

Response:

```
[
    {
        "extIn": 0,
        "updateFlag": 1,
        "relayOut": 0,
    }
]
```

4. Account-Oriented Query Interface

This section covers the resource representations for a specific account.

4.1 Transactions

Get transactions of all devices under a specific account starting from a specific time.

1) URL

https://[*server_url*]*/v2/api/transactions.json*

2) Response Content Type

JSON

3) HTTP Method

GET/POST

4) HTTP Parameters

Parameter	Required	Type	Description
key	true	string	Access key v2, assigned by system, login Partner page to see
startTime	true	String	ISO8601 format, yyyy-MM-ddTHH:mm:ssZ The server will return the transactions with a createTime later than the startTime, aka, transactions received by the server after the startTime.
count	true	int	Maximum number of transactions requested Default: 1000 Ranges: 1-10000

5) HTTP Body

{}

6) HTTP Response

A JSON array of device info objects with field definitions as below:

Field	Type	Length	Description
deviceId	String	14	Device ID
battery	float		Battery level in voltage
signals	int		0-100%
lat	float		latitude, six decimal places
lng	float		longitude, six decimal places
heading	int		in degree, 0-360, 0 means North
alt	int		altitude, in meters
speed	float		in km/h
reportTime	string		latest location update time in UTC eg,yyyy-MM-ddTHH:mm:ssZ
createTime	string		When the server received the data (UTC) eg,yyyy-MM-ddTHH:mm:ssZ
fwVer	int		firmware version. it may be null.
blVer	int		boot loader version, it may be null
charging	int		0 not charging, 1 charging
event	int		NULL, no data, 0 idle, 1 driving, 2 ACC off
totalMileage	int		Total mileage in meters;
updateIndex	int		index of a trail point
gpsState	Int		Location status: 0 not located, 1 2 3 located
hdp	int		Horizontal accuracy -1 indicates no record
accuracy	float		positional accuracy
country	String		address
state	String		About the address
city	String		city
streetAddress	String		Street Address
zip	String		Detailed Address
temp	int		Temperature
acc	int		Whether the transmitter is running, 0 is not running, 1 is running
voltage	float		Residual Voltage
solar	int		Solar charging current
current	int		charge current

7) Example

Request:

Method: POST
URL: https://api.anytrek.com/v2/api/transactions.json
Parameters: key=c0d5f837524a44b9a030736ec1b9eb63&startTime=2019-11- 04T07:00:00-0000&count=400
Body: {}

Response:


```
[{
"deviceId": 86610402349293,
"gpsState": 1,
"lat": 33.77822,
"lng": -118.24782,
"hdp": null,
"heading": 0,
"speed": 0,
"accuracy": 0.0,
"alt": 0, "country": null, "state": null,
"city": null,
"streetAddress": null, "zip": null, "totalMileage": 67617887,
"temp": 0,
"charging": 1,
"acc": 0,
"battery": 100, "voltage": null, "signals": 90,
"solar": null,
"event": 0,
"current": 315,
"fwVer": 48,
"blVer": 4,
"reportTime": "2019-11-04T07:57:19+0000", "createTime": "2019-11-04T07:57:20+0000",
"updateIndex": 19320,
}]
```


5. Error Response

1) Response Content Type

JSON

2) Response Body

A JSON object of error status, Ex: {"errorCode":200,"status":"error"}

Field	Type	Length	Description
status	string		"error"
errorCode	int		Error code, see Error Code description below

ErrorCode

Error Code	Description
1	Unknown parameters
2	Parameter error
100	imei or password is empty
101	IMEI or TrackerId is empty
102	Tracker not found based on IMEI or TrackerID
505	phone number is empty

6. Precautions

The request parameter with "+" is replaced with "%2B"