verizon

ThingSpace

Global Asset Tracker

Streaming Quick Start Guide

Contents

Overview

This document will lead you through the steps needed to set up a Verizon Global Asset Trackerto stream data to a target endpoint, such as a visualization platform like Freeboard or an account with a Cloud Service Provider like AWS.

The high-level steps are listed here. Most of them are done by sending API requests to thingspace.verizon.com.

1. Charge the Global Asset Tracker.
2. Use the credentials provided by your Verizon account representative to obtain the authentication and authorization tokens needed to send API requests.
3. Configure the Global Asset Tracker to your desired reporting frequency.
4. Tell ThingSpace where to send the device data.
5. Configure your target endpoint to receive the data.

Unpack and power on Global Asset Tracker

After unpacking the Global Asset Tracker, use the provided USB wall charger to plug it in to charge. The green Power LED will flash while the Sensor is charging, and will stay on when it is fully charged, which can take up to 4 hours.

NOTE: Only use the provided USB wall charger to charge the Global Asset Tracker. The device USB port does not provide access to data or configuration functions.

Obtain a ThingSpace OAuth2 session token

Your application must send an API token in the header of every API request. We use the OAuth2 “client credentials” grant type, and we require that the application key and secret are Base64 encoded. To obtain an API token:

1. Sign in to thingspace.verizon.com. (Sign Up for credentials if you have not previously signed up for a ThingSpace account. Also note this is different from UWS Account credentials).
2. Click your name in the upper right corner of the page, then select **Account settings** and click **Key Management**.
3. Copy the key and the secret and store them in a secure place for your application to use. If you do not see a key and secret, click on “Add Key Set” to obtain a key and secret.
4. Concatenate the key and the secret, with a colon between them into a continuous string, like this: `my_key_value:my_secret_value`
5. Encode the entire string in Base64 format. (To learn more about encoding in Base64 format, visit https://www.base64encode.org/.)
6. Send a POST request to https://thingspace.verizon.com/api/ts/v1/oauth2/token with the encoded string in the header. See example of the POST below.

The response will contain a token that you must include in the header of all API requests. **The token will be valid for one hour from when it was first issued**, and any further token requests during that hour will return the same token.

Sample request and response

Request Example

```
curl -X POST \
  https://thingspace.verizon.com/api/ts/v1/oauth2/token \
  -H 'authorization: Basic UW1PZHV6V3N2bTdNSHZoSETp5QUtGUk1ZWFhna2lTZ0hzV1U2YmlpbEZSUH' \
  -H 'content-type: application/x-www-form-urlencoded' \
  -d grant_type=client_credentials
```

Response Example

```
{
  "access_token": "d13c10f356e65b36ad5ea89d86f12247",
  "scope": "am_application_scope default",
  "token_type": "Bearer",
  "expires_in": 3600
}
```

Use UWS credentials to get a VZ-M2M-Token

Your application must send a VZ-M2M-Token in the header of every request to show that it is authorized to access the devices and other resources in your account. Your Verizon account representative can provide UWS credentials for use during the pilot:

- UWS Username
- UWS Password
- Account ID

Use the UWS username and password in the body of a POST request to https://thingspace.verizon.com/api/m2m/v1/session/login. Use the ThingSpace OAuth session token from step 2 as the authorization bearer value in the header of the request.

Sample request and response

Request Example

```
curl -X POST https://thingspace.verizon.com/api/m2m/v1/session/login \
  -H 'authorization: Bearer d13c10f356e65b36ad5ea89d86f12247' \
  -H 'content-type: application/json' \
  -d '{
    "username": "testuser",
    "password": "testpassword"
}'
```

Response Example

```
{
  "sessionToken": "32a61f4b-575a-4204-92d2-9460891fceaf"
}
```

Use the sessionToken value as the VZ-M2M-Token value in the header of all subsequent requests.

NOTE: The sessionToken will remain valid as long as your application continues to use it, but it will expire after 20 minutes of inactivity.

Change UWS password

You can request a new, random UWS password programmatically by sending a request to reset the password.

Verify devices (optional)

The devices should arrive already activated on the network and added to the account that you will use. You can verify the device status using your billing account ID attained from your account representative. This step will assure that everything is in the system and ready for use.

Sample request and response

Request Example

```
curl -X POST https://thingspace.verizon.com/api/cc/v1/devices/actions/query \
  -H 'authorization: Bearer d13c10f356e65b36ad5ea89d86f12247' \
  -H 'vz-m2m-token: 32a61f4b-575a-4204-92d2-9460891fceaf' \
  -H 'content-type: application/json' \
  -d '{
    "accountidentifier": {"billingaccountid":"1223334444-00001"},
    "$selection": {"billingaccountid":"1223334444-00001"}
}'
```

Response Example

```
[
  {
    "actionstate": {
      "configuration": {
        "configuration": {
          "Date": "2018-12-29T00:12:40.230729962Z",
          "State": "update"
        }
      }
    },
    "billingaccountid": "1223334444-00001",
    "createdon": "2018-12-21T00:28:43.013Z",
    "eventretention": 90,
    "iccid": "<your device iccid>",
    "id": "<your ThingSpace Device Unique id>",
    "imei": <your device IMEI>,
    "kind": "ts.device.cHeAssetTracker",
    "lastupdated": "2018-12-29T00:12:40.231Z",
    "providerid": "9dfcfa69-a1c8-4eae-8611-b282646bb113",
    "qrcode": "<your device IMEI>",
    "refid": "<your device IMEI>",
    "refidtype": "imei",
    "state": "ready",
```

```
    "version": "1.0",
    "versionid": "7465d5b6-0afe-11e9-b78b-02420a46140b"
  },
```


NOTE: The response will contain the above information for each of the devices in the account.

Configure the device reporting period

There are two device configuration options:

- frequency - how frequently the sensors are read and reported
- location_mode - whether GPS is enabled or disabled

These are the valid values for "frequency" and "location_mode". Note that the values are case sensitive.

"frequency" value	Sensor read and reporting frequency
High	15 min
Medium	1 hour
Low	4 hours
Ultra Low	24 hours

"location_mode" value	GPS functionality
gps	GPS enabled. Location is reported with other sensor values per frequency settings. **NOTE:** GPS reception may not work well indoors. When GPS signal reception is not available the device reports coarse location. See Location below.
disable	GPS disabled

Sample request and response

This sample request sets the frequency to High and enables the GPS functionality. The device is specified by its IMEI.

Example Request

```
curl -X POST \
  https://thingspace.verizon.com/api/cc/v1/devices/configuration/actions/set \
  -H 'authorization: Bearer d13c10f356e65b36ad5ea89d86f12247' \
  -H 'vz-m2m-token: 32a61f4b-575a-4204-92d2-9460891fceaf' \
  -H 'content-type: application/json' \
  -d '{
    "accountidentifier":{"billingaccountid": "1223334444-00001"},
    "resourceidentifier":{"imei":<device IMEI>},
    "configuration":{"frequency":"High","location_mode":"gps"}
```

```
}'
```

Example Response:


```
[
  {
    "action": "set",
    "createdon": "2019-01-18T07:10:21.601143024Z",
    "deviceid": "0431f44e-5ae2-6c30-f75f-9f0ea29c2574",
    "fields": {
      "configuration": {
        "frequency": "High",
        "location_mode": "gps"
      }
    },
    "foreignid": "e1f15861-7de7-69cb-ed7d-b4a92e091bc4",
    "id": "85a1351a-650e-6abf-f939-c8f5cd4e64aa",
    "kind": "ts.event.configuration",
    "lastupdated": "2019-01-18T07:10:21.601143337Z",
    "name": "SetConfigurationReq",
    "state": "pending",
    "transactionid": "4a7b2e59-2993-409b-8794-ca65cb27eabf",
    "version": "1.0"
  }
]
```


NOTE: ThingSpace cannot send configuration change requests immediately because Global Asset Trackers sleep during the "frequency" period and are not reachable until the specified interval elapses. When the frequency period has elapsed, the device wakes up to read sensors and send data to ThingSpace, and ThingSpace is able to send configuration changes to the device. The device then goes back to sleep for the period of time set by the (new) configuration.

Once you send a POST /devices/actions/configuration/set request for a device, you cannot cancel or overwrite it. After the device wakes up and gets the configuration change request, you can send another request, if needed, and that request will be queued for when the device wakes up again.

Location

Global Asset Tracker firmware version v2.2.0 and higher supports CellId based coarse location reporting. Your device is configured to report location by default and attempts to provide GPS/GNSS based precise location. When your device is unable to provide GPS based location it will use CellId to provide coarse location.

Location reporting via API

The API request /cc/v1/devices/fields/locations/actions/history can be used to obtain location information of a Global Asset Tracker device. Below are example responses to the API when

i. API response contains precise location based on GPS/GNSS

ii. API response contains coarse location based on CellId

Location Response to API based on GPS/GNSS


```
{
       "action":"update",
       "createdon":"2019-05-17T22:40:01Z",
       "deviceid":"24d1eb09-fe77-6f95-f2e8-1fca8e2c50ce",
       "fields":{
          "location":{
             "accuracy":"9.0",
             "latitude":"32.8977737",
             "longitude":"-117.2027053",
             "type":"gps"
          }
       },
       "id":"e5c1d620-879a-6a00-e7ad-b7c4b5e7905d",
       "kind":"ts.event",
       "lastupdated":"2019-05-17T22:40:48.343974582Z",
       "state":"update",
       "transactionid":"864508030136201-OnBoard-7cbae151-089e-498e-99a7-632b3f7e664d",
       "version":"1.0",
       "versionid":"d0e39483-78f4-11e9-a6b5-02420a491c0c"
   }
```


Coarse Location Response to API based on CellId


```
{
    "action":"update",
    "createdon":"2019-05-17T21:40:01Z",
    "deviceid":"24d1eb09-fe77-6f95-f2e8-1fca8e2c50ce",
    "fields":{
       "location":{
          "altitude":76.69,
          "latitude":32.905369,
          "longitude":-117.192574,
```

```
            "radius":1598.64,
            "type":"cellid"
         }
      },
      "id":"e591ee05-bd36-6b0c-fc74-4bff20b2da1d",
      "kind":"ts.event",
      "lastupdated":"2019-05-17T21:41:12.9Z",
      "state":"update",
      "transactionid":"864508030136201-OnBoard-cbe9148f-c980-4b18-bb76-e626a031c2a0",
      "version":"1.0",
      "versionid":"7dc23eeb-78ec-11e9-b733-02420a581e0e"
}
```

Alarms

Use the API to configure threshold values of sensor data to enable alarms on your device.

Setting alarms

Set an alarm configuration on the device with threshold values of sensor data. The device alarm will be triggered in the form of an event when the sensor values on the device reach the threshold values in the alarm configuration.

API to configure

```
curl -k -v -X POST https://thingspace.verizon.com/api/cc/v1/devices/actions/set \
  -H "Content-Type: application/json" \
  -H 'Authorization: Bearer <<Oauth 2.0 Token>>' \
  -H 'VZ-M2M-Token: <<VZ-M2M-Token>>'\
  -d '{
   "accountidentifier":{"billingaccountid":"<<Billing Account Id>>"},
   "resourceidentifier":{"imei":<<IMEI of Device>>},
   "deviceConfig": {"<<Alarm Key Word>>":{"alarmType":4,
   "threshold":"<<Lower Bound Value>>","thresholdRange":"<<Highest Bound Value>>"
}}}
```

The curl above will enable the alarm configuration on top of the device. Add a value from the list below in between the (**<< >>**).

- **Billing Account Id** - configured on the Global Asset Tracker device.
- **Oauth 2.0 Token-** authorizes device.
- **VZ-M2M-Token**- authorizes device.
- **IMEI of Device** - IMEI of the device.
- **Alarm Key Word** - Each sensor type has its own Alarm Key Word. See below for a list of key words corresponding to each sensor type.

Sensor Type	Alarm Key Word
Temperature	tempAlarm
Light	liteAlarm
Humidity	humiAlarm
Pressure	presAlarm

Sensor Type	Alarm Key Word
battery	battAlarm

- **Lower Bound Value** - Lower Bound of sensor value for which the alarm is to be triggered.
- **Higher Bound Value** - Higher Bound of sensor value for which the alarm is to be triggered.

Example Requests and Responses

Temperature

Here is an example on how to configure a temperature alarm on a device with the threshold of (15 – 20C). The device will send out alarms when the temperature is below 15C and above 20C.

Example request

```
curl -k -v -X POST https://thingspace.verizon.com/api/cc/v1/devices/actions/set \
  -H "Content-Type: application/json" \
  -H 'Authorization: Bearer 18261b8341d07fc32c760c2a5acaad27' \
  -H 'VZ-M2M-Token: 885b3ba5-b149-4fe8-91c5-51efb1b7462a' \
  -d'{
   "accountidentifier":{"billingaccountid":"123456789-00001"},
   "resourceidentifier":{"imei":864508030123456},
   "deviceConfig": {"tempAlarm":{"alarmType":4,"threshold":"15","thresholdRange":"20"
}}}'
```

Example response

```
[{
   "action":"set",
   "capabilityid":"00000000-0001-0001-0001-000000000000",
   "createdon":"2019-08-08T17:14:43.276220009Z",
   "deviceid":"04314840-7e38-6e64-e6e7-cfd1f0be5515",
   "fields":{"deviceConfig":{"tempAlarm":{"alarmType":4,
   "threshold":"15","thresholdRange":"20"}}},
   "id":"a5a1935d-222d-6ecf-efa4-18be00ab5a7c",
   "kind":"ts.event","lastupdated":"2019-08-08T17:14:43.276220072Z",
   "state":"pending",
   "transactionid":"91a6868e-f593-4102-a12f-ed8319c5909a",
   "version":"1.0","versionid":"037c6952-ba00-11e9-94f6-02420a4e1a12"
}]
```

Pressure

Here is an example of how to configure a pressure alarm on a device with the threshold of (600 – 710). The device will send out alarms when pressure is below 600 and above 710.

Example request

```
curl -k -v -X POST https://thingspace.verizon.com/api/cc/v1/devices/actions/set \
  -H "Content-Type: application/json" \
  -H 'Authorization: Bearer 18261b8341d07fc32c760c2a5acaad27' \
  -H 'VZ-M2M-Token: 885b3ba5-b149-4fe8-91c5-51efb1b7462a' \
  -d'{
   "accountidentifier":{"billingaccountid":"123456789-00001"},
   "resourceidentifier":{"imei":864508030123456},
   "deviceConfig": {"presAlarm":{"alarmType":4,"threshold":"600","thresholdRange":"710"}}
}'
```

Example response

```
[{
  "action":"set",
  "capabilityid":"00000000-0001-0001-0001-000000000000",
  "createdon":"2019-08-08T17:21:12.321021069Z",
  "deviceid":"04314840-7e38-6e64-e6e7cfd1f0be5515",
  "fields":{"deviceConfig":{"presAlarm":{"alarmType":4,"threshold":"600",
  "thresholdRange":"710"}}},
  "id":"0541c33e-6415-68ad-e21f-3f9450c4c707",
  "kind":"ts.event","lastupdated":"2019-08-08T17:21:12.321021139Z",
  "state":"pending",
  "transactionid":"d203ee1d-dfa8-4156-a3ef-1339cdfd80ec",
  "version":"1.0","versionid":"eb5fefdc-ba00-11e9-9576-02420a4e1a12"
}]
```

Light

Here is an example of how to configure a light alarm on a device with the threshold of (100 – 110). The device will send out alarms when light is below 100 and above 110.

Example request

```
curl -k -v -X POST https://thingspace.verizon.com/api/cc/v1/devices/actions/set \
  -H "Content-Type: application/json" \
  -H 'Authorization: Bearer 18261b8341d07fc32c760c2a5acaad27' \
  -H 'VZ-M2M-Token: 885b3ba5-b149-4fe8-91c5-51efb1b7462a' \
  -d '{
   "accountidentifier":{"billingaccountid":"123456789-00001"},\
```

```
   "resourceidentifier":{"imei":864508030123456},
   "deviceConfig":{"liteAlarm":{"alarmType":4,"threshold":"100","thresholdRange":"110"}}
}'
```

Example response

```
[{
  "action":"set","capabilityid":"00000000-0001-0001-0001-000000000000",
  "createdon":"2019-08-08T17:17:29.461331282Z",
  "deviceid":"04314840-7e38-6e64-e6e7-cfd1f0be5515",
  "fields":{"deviceConfig":{"liteAlarm":{"alarmType":4,"threshold":"100",
  "thresholdRange":"110"}}},
  "id":"e5116a42-0e28-6631-e830-8fdc180a85ac",
  "kind":"ts.event","lastupdated":"2019-08-08T17:17:29.461331332Z",
  "state":"pending",
  "transactionid":"c6f47f25-efb3-4be4-9cd6-4ffd69af07cf",
  "version":"1.0","versionid":"668a3f3b-ba00-11e9-94fe-02420a4e1a12"
}]
```

Humidity

Here is an example on how to configure a humidity alar om a device with the threshold of (50– 60. The devcie will send out alarms with humidity is below 50 and above 60.

Example alarm for humidity sensor

```
curl -k -v -X POST https://thingspace.verizon.com/api/cc/v1/devices/actions/set \
  -H "Content-Type: application/json" \
  -H 'Authorization: Bearer 18261b8341d07fc32c760c2a5acaad27' \
  -H 'VZ-M2M-Token: 885b3ba5-b149-4fe8-91c5-51efb1b7462a' \
  -d'{
   "accountidentifier":{"billingaccountid":"123456789-00001"},
   "resourceidentifier":{"imei":864508030123456},
   "deviceConfig": {"humiAlarm":{"alarmType":4,"threshold":"50","thresholdRange":"60"}}
}'
```

Example response

```
[{
  "action":"set",
  "capabilityid":"00000000-0001-0001-0001-000000000000",
  "createdon":"2019-08-08T17:20:04.91388507Z",
  "deviceid":"04314840-7e38-6e64-e6e7-cfd1f0be5515",
  "fields":{"deviceConfig":{"humiAlarm":{"alarmType":4,"threshold":"50",
  "thresholdRange":"60"}}},
  "id":"c54194ee-bb41-64ba-fc7a-eef2070caa8e",
```

```
  "kind":"ts.event",
  "lastupdated":"2019-08-08T17:20:04.913885138Z",
  "state":"pending",
  "transactionid":"ccce3cfb-b60a-425d-a12c-0efa8a6a3c26",
  "version":"1.0",
  "versionid":"c3326c9d-ba00-11e9-9575-02420a4e1a12"
}]
```


Battery

Here is an example of how to configure a battery alarm on a device with the threshold of (90 – 100). The device will send out alarms when battery is below 90 and above 100.

Example request

```
curl -k -v -X POST https://thingspace.verizon.com/api/cc/v1/devices/actions/set \
  -H "Content-Type: application/json" \
  -H 'Authorization: Bearer 18261b8341d07fc32c760c2a5acaad27' \
  -H 'VZ-M2M-Token: 885b3ba5-b149-4fe8-91c5-51efb1b7462a' \
  -d'{
   "accountidentifier":{"billingaccountid":"123456789-00001"},
   "resourceidentifier":{"imei":864508030123456},
   "deviceConfig": {"battAlarm":{"alarmType":4,"threshold":"90","thresholdRange":"100"}}
}'
```

Example response


```
 [{
"action":"set",
"capabilityid":"00000000-0001-0001-0001-000000000000",
"createdon":"2019-08-08T17:22:44.493371697Z",
"deviceid":"04314840-7e38-6e64-e6e7-
cfd1f0be5515","fields":{"deviceConfig":{"battAlarm":{"alarmType":4,
"threshold":"90","thresholdRange":"100"}}},
"id":"25816a60-d18e-6c8b-fb1c-f674b91c61ef",
"kind":"ts.event","lastupdated":"2019-08-08T17:22:44.493371922Z",
"state":"pending",
"transactionid":"2267b343-8e69-4fd7-ab4b-5d06c97b6288",
"version":"1.0",
"versionid":"2250524f-ba01-11e9-9577-02420a4e1a12"
}]
```

Alarm reporting examples

Once the Alarm is set on a device using an API call the device will start sending alarms whenever the sensor value crosses the threshold and threshold range. The alarms are events on the device. Sample alarms on the device are provided below.

Temperature alarm

```
deviceAlarm{{"temperature":{"Threshold":"15","ThresholdRange":"20","alarmType":"4","senso
rReading":"24","state":1}}}
```

Pressure

```
deviceAlarm{{"pressure":{"Threshold":"600","ThresholdRange":"710","alarmType":"4","sensor
Reading":"800", "state":1}}
```

Light alarm

```
deviceAlarm{{"light":{"Threshold":"100","ThresholdRange":"110","alarmType":"4","sensor
Reading":"98", "state":1}}
```

Humidity alarm

```
deviceAlarm{{"humidity":{"Threshold":"50","ThresholdRange":"60","alarmType":"4","sensorRe
ading":"98", "state": 1}}
```

Battery alarm

```
deviceAlarm{{"battery":{"Threshold":"90","ThresholdRange":"100","alarmType":"4","sensorRe
ading":"88","state":1}}
```

Disabling alarms example requests

Disabling temperature alarm

```
curl -k -v -X POST https://thingspace.verizon.com/api/cc/v1/devices/actions/set \
 -H "Content-Type: application/json" \
 -d'{
   "accountidentifier":{"billingaccountid":"123456789-00001"},
   "resourceidentifier":{"imei":864508030123456},
   "deviceConfig": {"tempAlarm":{"alarmType":0,"threshold":"20","thresholdRange":"25"}}
}'
```

Disabling pressure alarm

```
curl -k -v -X POST https://thingspace.verizon.com/api/cc/v1/devices/actions/set \
  -H "Content-Type: application/json" \
  -H 'Authorization: Bearer 18261b8341d07fc32c760c2a5acaad27' \
```

```
  -H 'VZ-M2M-Token: 885b3ba5-b149-4fe8-91c5-51efb1b7462a' \
  -d'{
   "accountidentifier":{"billingaccountid":"123456789-00001"},
   "resourceidentifier":{"imei":864508030123456},
   "deviceConfig": {"presAlarm":{"alarmType":0,"threshold":"600","thresholdRange":"710"}}
}'
```

Disabling light alarm

```
curl -k -v -X POST https://thingspace.verizon.com/api/cc/v1/devices/actions/set \
  -H "Content-Type: application/json" \
  -H 'Authorization: Bearer 18261b8341d07fc32c760c2a5acaad27' \
  -H 'VZ-M2M-Token: 885b3ba5-b149-4fe8-91c5-51efb1b7462a' \
  -d'{
   "accountidentifier":{"billingaccountid":"123456789-00001"},
   "resourceidentifier":{"imei":864508030123456},
   "deviceConfig": {"liteAlarm":{"alarmType":0,"threshold":"100","thresholdRange":"110"}}
}'
```

Disabling humidity alarm

```
curl -k -v -X POST https://thingspace.verizon.com/api/cc/v1/devices/actions/set \
  -H "Content-Type: application/json" \
  -H 'Authorization: Bearer 18261b8341d07fc32c760c2a5acaad27' \
  -H 'VZ-M2M-Token: 885b3ba5-b149-4fe8-91c5-51efb1b7462a' \
  -d'{
   "accountidentifier":{"billingaccountid":"123456789-00001"},
   "resourceidentifier":{"imei":864508030123456},
   "deviceConfig": {"humiAlarm":{"alarmType":0,"threshold":"50","thresholdRange":"60"}}
}'
```

Disabling battery alarm

```
curl -k -v -X POST https://thingspace.verizon.com/api/cc/v1/devices/actions/set \
  -H "Content-Type: application/json" \
  -H 'Authorization: Bearer 18261b8341d07fc32c760c2a5acaad27' \
  -H 'VZ-M2M-Token: 885b3ba5-b149-4fe8-91c5-51efb1b7462a' \
  -d'{
   "accountidentifier":{"billingaccountid":"123456789-00001"},
   "resourceidentifier":{"imei":864508030123456},
   "deviceConfig": {"battAlarm":{"alarmType":0,"threshold":"90","thresholdRange":"100"}}
}'
```

Set up streaming

You can stream data from ThingSpace to various endpoints, such as to accounts with cloud service providers. This section first describes how to set up generic streaming, and then how to configure ThingSpace to stream to an AWS account.

Streaming to a host:port endpoint

Streaming requires a target resource to define the endpoint, and a subscription resource to define what is streamed to the target.

Create a target to define the streaming endpoint

A target resource defines an endpoint that can be used for streaming. After creating a target, use the target ID from the response when you create a subscription to set up a data stream.

Use the POST /targets request to create a target, with these parameters:

- accountidentifier.**billingaccountid** (required) The ID of the billing account that is making the request, in the format 1234567890-12345.
- **name** (optional) Name of the target.
- **description** (optional) Descriptive information about the target.
- **addressscheme** (required) The transport format. Valid values for streaming are:
  - streamrest - streamed REST data to a defined endpoint
  - streamawsiot - streamed data to an AWS account
- **address** (required) The endpoint for data streams. The format depends on the selected `addressscheme`, but is often a host:port value. The endpoint must support a HTTP over TLS (HTTPS) connection and the endpoint server TLS certificate must be issued by a trusted certificate authority.
- **Authorization Schemes supported – the following authorization schemes are supported, use any one**
  - No Authorization
    - No extra fields have to be included in the body of the request
  - Basic Authorization (HTTP Headers)
    - Add the following field to the body of the request

      “httpheaders” : { "Authorization": "Basic <<<Encoded with Base64>>>" }
  - OAuth2
    - Add the following field to the body of the request

      "key1": "Bearer <<<OAuth token>>>"

- OAuth2 with Refresh Token
  - Add the following fields to the body of the request

```
"key1": "Bearer <<<OAUTH_TOKEN>>>",
 "oauth": {
 "body": {
         "grant_type": "refresh_token",
         "refresh_token": "<<<REFRESH_TOKEN>>>",
         "scope": "<<<SCOPE_OPTIONAL>>>"
        },
 "headers":{
     "Authorization": "Basic <<<BASE64_CLIENTID:CLIENTSECRET>>>",
     "Content-Type": "application/x-www-form-urlencoded"
       },
   "host":{
      "hostandpath": "<<<REFRESH_TOKEN_URL>>>"
      }
   }
```

NOTE: To obtain the BASE64_CLIENTID:CLIENTSECRET do the following.

iii. Concatenate the CLIENTID and the CLIENTSECRET, with a colon between them into a continuous string, like this: CLIENTID:CLIENTSECRET.

iv. Encode the entire string in Base64 format. (To learn more about encoding in Base64 format, visit https://www.base64encode.org/).

v. Use the Base64 encoded value of CLIENTID:CLIENTSECRET in the API.

Sample request and response

Request Example

```
curl -X POST https://thingspace.verizon.com/api/cc/v1/targets \
  -H 'authorization: Bearer d13c10f356e65b36ad5ea89d86f12247' \
  -H 'vz-m2m-token: 32a61f4b-575a-4204-92d2-9460891fceaf' \
  -H 'content-type: application/json' \
  -d '{
    "accountidentifier":{"billingaccountid": "1223334444-00001"},
    "billingaccountid":"1223334444-00001",
    "address":"https://www.test.com:1536",
    "addressscheme":"streamrest",
```

```
    "name":"test",
    "description":"description"
}'
```

Response Example

```
{
  "address": "https://www.test.com:1536",
  "addressscheme": "streamrest",
  "billingaccountid":"1223334444-00001",
  "createdon": "2018-12-21T04:37:42.651Z",
  "id": "0e411230-c3eb-64dc-f5f4-1020364aa81f",
  "kind": "ts.target",
  "lastupdated": "2018-12-21T04:37:42.651Z",
  "version": "1.0",
  "versionid": "27aca5a4-04da-11e9-bff3-02420a5e1b0b"
}
```

You will need the highlighted ID of the target when you create a subscription. Save the Target id value.

Create a subscription

A subscription defines a streaming channel that sends data from devices in the account to an endpoint defined in a target resource. Note the requirements for these values:

- **email** is the address to which any error reports will be delivered.
- **streamkind** must be ts.event to stream all device data.
- **targetid** must be the ID of the target that was created to define the endpoint.
- **name** is not required, but can be used when querying for the subscription later.

Sample request and response

Request example

```
curl -X POST https://thingspace.verizon.com/api/cc/v1/subscriptions \
  -H 'authorization: Bearer d13c10f356e65b36ad5ea89d86f12247' \
  -H 'vz-m2m-token: 32a61f4b-575a-4204-92d2-9460891fceaf' \
  -H 'content-type: application/json' \
  -d '{
    "accountidentifier":{"billingaccountid":"1223334444-00001"},
    "billingaccountid":"1223334444-00001",
    "email":"me@mycompany.com",
    "streamkind":"ts.event",
    "targetid":"0e411230-c3eb-64dc-f5f4-1020364aa81f",
    "name":"subscription1"
```

```
}'
```

Example response:

```
{
  "configurationfailures": 0,
  "billingaccountid":"1223334444-00001",
  "createdon": "2019-01-28T03:18:34.323Z",
  "email": "me@mycompany.com",
  "id": "b8719618-bf7e-6b0a-e40a-5939d00b17a1",
  "kind": "ts.subscription",
  "laststreamingstatus": "",
  "laststreamingtime": "0001-01-01T00:00:00Z",
  "lastupdated": "2019-01-28T03:18:34.323Z",
  "name": "subscription1",
  "networkfailures": 0,
  "streamfailures": 0,
  "streamkind": "ts.event",
  "targetid": "0e411230-c3eb-64dc-f5f4-1020364aa81f",
  "version": "1.0",
  "versionid": "6525b340-22ab-11e9-ac5d-02420a40130c"
}
```

Save the subscription id created, in case you need to create a new subscription you will have to delete this one.

Streaming to AWS

Generate an External ID

ThingSpace uses an external ID for increased security when streaming to AWS. You generate the ID in ThingSpace, then use it when configuring an AWS account and a ThingSpace target resource.

Sample request and response

Example request

```
curl -X POST https://thingspace.verizon.com/api/cc/v1/targets/actions/newextid \
  -H 'authorization: Bearer d13c10f356e65b36ad5ea89d86f12247' \
  -H 'vz-m2m-token: 32a61f4b-575a-4204-92d2-9460891fceaf' \
  -H 'content-type: application/json' \
  -d '{"accountidentifier": {"billingaccountid":"1223334444-00001"}}'
```

Example response

```
{
  "externalid": "XsosZrEEkfPuR3aGOk2iHIr6tXN275ioJF6bezIrQB9EbzpTRep8J7RmV7HQ=="
```

```
}
```

Configure AWS Account

Follow these steps to allow ThingSpace to send device data to your AWS account.

1. Sign in to AWS.
2. Browse to **IAM** (Identity and Access Management).
3. From the IAM Dashboard, click **Roles** in the left column.
4. Click **Create role**.
5. Select **Another AWS account** as the type of trusted identity.
6. Enter the Verizon **Account ID**, which is **675479154635**.
7. Check the box to **Require external ID** and paste in the ID generated in section 6.2.1.
8. Click **Next: Permissions**.
9. Select these permissions:
   - AWSIotDataAccess
   - AWSIoTFullAccess
   - AWSIoTThingsRegistration
10. Click **Next: Tags**.
11. No AWS tags are required. Click **Next: Review**.
12. Enter a **name** for the role (for example, "ThingSpace").
13. Click **Create Role**.

AWS will display the role summary page. You will need the Role **ARN** from that page when you create a target in ThingSpace.

Create a target for AWS streaming

A target resource defines an endpoint that can be used for streaming. After creating a target, use the target ID from the response when you create a subscription to set up a data stream.

Note the requirements for these values to stream to AWS:

- **addressscheme** must be "streamawsiot".
- **address** is the ARN provided by AWS for the role created in the previous step.
- **region** is the AWS region where your application connects to AWS IoT services. See AWS Regions and Endpoints for a table of regions for the AWS IoT Core service. Note that Things and data from one region will not be visible in another region.

- **externalid** is the long string generated in section 6.2.1.
- **name** (and description) are not required, but resource names can be used to query for resources later.

Sample request and response

Request Example:

```
 curl -X POST \
  https://thingspace.verizon.com/api/cc/v1/targets \
  -H 'authorization: Bearer d13c10f356e65b36ad5ea89d86f12247' \
  -H 'vz-m2m-token: 32a61f4b-575a-4204-92d2-9460891fceaf' \
  -H 'content-type: application/json'
  -d '{
    "accountidentifier":{"billingaccountid": "1223334444-00001"},
    "billingaccountid":"1223334444-00001",
    "name":"AWS target",
    "description":"description",
    "addressscheme":"streamawsiot",
    "address":"arn:aws:iam::15625542789:role/ThingSpace",
    "externalid":"XsosZrEEkfPuR3aGOk2iHIr6tXN275ioJF6bezIrQB9EbzpTRep8J7RmV7HQ==",
    "region":"us-east-1"
}'
```

Response Example:

```
{
    "address": "arn:aws:iam::15625542789:role/ThingSpace",
    "addressscheme": "streamawsiot",
    "billingaccountid":"1223334444-00001",
    "createdon": "2019-01-28T02:51:54.941Z",
    "externalid": "XsosZrEEkfPuR3aGOk2iHIr6tXN275ioJF6bezIrQB9EbzpTRep8J7RmV7HQ==",
    "region":"us-east-1"
    "id": "ee113948-8f03-67e0-f42a-8a33cbb6ce34",
    "kind": "ts.target",
    "lastupdated": "2019-01-28T02:51:54.941Z",
    "name": "AWS target",
    "version": "1.0",
    "versionid": "abd77755-22a7-11e9-a2ee-02420a46120b"
}
```

You will need the highlighted ID of the target when you create a subscription.

Create a subscription

A subscription defines a streaming channel that sends data from devices in the account to an endpoint defined in a target resource. Note the requirements for these values:

- **billingaccountid** is the UWS billingaccountid provided.
- **email** is the address to which any error reports will be delivered.
- **streamkind** must be ts.event to stream all device data.
- **targetid** must be the ID of the target that was created to define the endpoint.
- **name** is not required, but can be used when querying for the subscription later.

Sample request and response

Request example

```
curl -X POST https://thingspace.verizon.com/api/cc/v1/subscriptions \
  -H 'authorization: Bearer d13c10f356e65b36ad5ea89d86f12247' \
  -H 'vz-m2m-token: 32a61f4b-575a-4204-92d2-9460891fceaf' \
  -H 'content-type: application/json' \
  -d '{
    "accountidentifier":{"billingaccountid":"1223334444-00001"},
    "billingaccountid":"1223334444-00001",
    "email":"me@mycompany.com",
    "streamkind":"ts.event",
    "targetid":"ee113948-8f03-67e0-f42a-8a33cbb6ce34",
    "name":"Account-level subscription"
}'
```

Example response:

```
{
  "configurationfailures": 0,
  "billingaccountid":"1223334444-00001",
  "createdon": "2019-01-28T03:18:34.323Z",
  "email": "me@mycompany.com",
  "id": "b8719618-bf7e-6b0a-e40a-5939d00b17a1",
  "kind": "ts.subscription",
  "laststreamingstatus": "",
  "laststreamingtime": "0001-01-01T00:00:00Z",
  "lastupdated": "2019-01-28T03:18:34.323Z",
  "name": "Account-level subscription",
  "networkfailures": 0,
  "streamfailures": 0,
```

```
  "streamkind": "ts.event",
  "targetid": "ee113948-8f03-67e0-f42a-8a33cbb6ce34",
  "version": "1.0",
  "versionid": "6525b340-22ab-11e9-ac5d-02420a40130c"
}
```

Save the subscription id created, in case you need to create a new subscription you will have to delete this one.

Viewing Data in AWS

After AWS and ThingSpace are set up as described on this page, ThingSpace registers devices in the account as "things" in your AWS account, and streams data from those devices to the AWS account.

- Go to the IoT Core page to see a summary of IoT traffic.
- Click Manage > Things to see things created by ThingSpace (and from other sources). The names of ThingSpace Things are a device identifier (such as the IMEI) prefaced by "ts_", such as ts_864508030084979.
- Click a Thing's name to see more information about that Thing.
- Click Shadow while viewing a Thing to see the latest data values from the Thing. Global Asset Tracker values will look similar to this:

```
{
  "reported": {
    "temperature": "20.7",
    "humidity": "36",
    "light": "0",
    "pressure": "896",
    "battery": "100",
    "acceleration": {
      "x": "0.0133",
      "y": "-1.0394",
      "z": "-0.0328"
    },
    "orientation": {
      "motionInX": "268.0",
      "motionInY": "0.0",
      "motionInZ": "0.0"
    },
    "signalStrength": "-58"
  }
}
```

Additional Cloud Connector API reference

Search for devices (search API) Post / Devices/ Actions / Search

Search for devices by property values. Returns an array of all matching device resources, including the latest sensor values and configuration settings for each device.

Request components

HTTP request

```
POST https://thingspace.verizon.com/api/cc/v1/devices/actions/search
```

Resource path and query parameters

None.

Header parameters

The request header must contain a current ThingSpace authorization token and a current VZ-M2M session token, and must set the content-type to JSON.

Property Name	Data Type	Description
Authorization required	string	HTTP Authorization request header containing a valid ThingSpace Bearer token.
VZ-M2M-Token required	string	A valid session token returned by a Connectivity Management API POST /session/login request using the account's UWS credentials.
Content-Type required	string	Must be `application/json`.

Request body

The request body specifies fields and values to match.

Property Name	Data Type	Description

accountidentifier required	JSON object	The ID of the authenticating billing account, in the format {"billingaccountid":"1234567890-12345"}.
$selection optional	JSON object	A comma-separated list of properties and comparator values to match against devices in the ThingSpace account. If the request does not include `$selection`, the response will include all devices to which the billingaccountid has access.

Example request body

Find a device with a specific ICCID

```
curl -X POST https://thingspace.verizon.com/api/cc/v1/devices/actions/search \
  -H 'Authorization: Bearer 16f6abbb55ce3980a82dd3a0c1d797af' \
  -H 'VZ-M2M-Token: dd483724-b734-4e13-b89a-15d254c310d9' \
  -H 'Content-Type: application/json' \
  -d '{
    "accountidentifier": {"billingaccountid":"1223334444-00001"},
    "$selection": {"iccid": "89148000003499233389"}
  }'
```

Find all device that were created after 12/31/2018

```
curl -X POST https://thingspace.verizon.com/api/cc/v1/devices/actions/search \
  -H 'Authorization: Bearer 16f6abbb55ce3980a82dd3a0c1d797af' \
  -H 'VZ-M2M-Token: dd483724-b734-4e13-b89a-15d254c310d9' \
  -H 'Content-Type: application/json' \
  -d '{
    "accountidentifier": {"billingaccountid":"1223334444-00001"},
    "$selection": {"createdon gt": "2018-12-31"}
  }'
```

Success Responses

Status 200

A success response includes an array of all matching devices. Each device includes the full device resource definition and the latest field (sensor) values.

Example Success Response

```
[
  {
    "actions": {
      "configuration": {
        "configuration": {
          "Date": "2019-02-14T02:34:53.149140091Z",
          "State": "update"
        }
```

```
      }
    },
    "billingaccountid": "1223334444-00001",
    "createdon": "2019-01-05T00:14:35.547Z",
    "eventretention": 90,
    "fields": {
      "acceleration": {
        "x": "0.0277",
        "y": "-1.0334",
        "z": "-0.0134"
      },
      "battery": "95",
        "light": {
          "Threshold": "100",
          "ThresholdRange": "21699",
          "alarmType": "4",
          "sensorReading": "98"
        }
      },
      "humidity": "29",
      "light": "150",
      "location": {
        "latitude": "37.2314796",
        "longitude": "-119.4692153"
      },
      "orientation": {
        "motionInX": "268.0",
        "motionInY": "1.0",
        "motionInZ": "0.0"
      },
      "pressure": "888",
      "signalStrength": "-58",
      "temperature": "19.2"
    },
    "iccid": "89148000003499233389",
    "id": "8461f530-2e31-6e87-e357-6c38251d41d0",
    "imei": 864508030084799,
    "kind": "ts.device.cHeAssetTracker",
    "lastupdated": "2019-02-17T03:57:26.577Z",
    "providerid": "9dfcfa69-a1c8-4eae-8611-b282646bb113",
    "qrcode": "864508030084799",
    "refid": "864508030084799",
    "refidtype": "imei",
    "services": [
      "configuration"
    ],
    "state": "ready",
    "tagids": [
      "4d110e4f-7a7c-6b26-eaac-31cc34c6e1d4",
      "cd81ed16-18ae-6c7d-eaba-2883b0395387"
    ],
    "version": "1.0",
    "versionid": "238a7e2e-3268-11e9-91d6-02420a472210"
```

```
  },
  {
    "actions": {
      "configuration": {
        "configuration": {
          "Date": "2019-01-11T03:04:49.435931518Z",
          "State": "update"
        }
      }
    },
    "billingaccountid": "1223334444-00001",
    "createdon": "2019-01-09T00:00:48.387Z",
    "eventretention": 90,
    "fields": {
      "acceleration": {
        "x": "-0.0762",
        "y": "-0.0152",
        "z": "1.0864"
      },
      "battery": "1",
      "humidity": "38",
      "light": "0",
      "location": {
        "latitude": "32.8980026",
        "longitude": "-117.2033309"
      },
      "orientation": {
        "motionInX": "0.0",
        "motionInY": "4.0",
        "motionInZ": "0.0"
      },
      "pressure": "1012",
      "signalStrength": "-86",
      "temperature": "20.2"
    },
    "iccid": "89148000003499234933",
    "id": "44314312-3cbc-651d-fec4-286e0baa5182",
    "imei": 864508030027310,
    "kind": "ts.device.cHeAssetTracker",
    "lastupdated": "2019-01-11T03:04:49.436Z",
    "providerid": "9dfcfa69-a1c8-4eae-8611-b282646bb113",
    "qrcode": "864508030027310",
    "refid": "864508030027310",
    "refidtype": "imei",
    "services": [
      "configuration"
    ],
    "state": "ready",
    "version": "1.0",
    "versionid": "a8743c8a-154d-11e9-ba53-02420a401605"
  }
]
```

Failure Responses

Status 4xx

All error messages are returned in this format:

```
{
  "error": "error code string",
  "error_description": "error message string",
  "cause": "further explanation"
}
```

Search device fields

POST /devices/fields/actions/history/search

Search device event history to find events that match criteria. Sensor readings, configuration changes, and other device data are all stored as events; use this request to search all device data.

Request Components

HTTP Request

```
POST https://thingspace.verizon.com/api/cc/v1/devices/fields/actions/history/search
```

Resource Path and Query Parameters

None.

Header Parameters

The request header must contain a current ThingSpace authorization token and a current VZ-M2M session token, and must set the content-type to JSON.

Property Name	Data Type	Description
Authorization required	string	HTTP Authorization request header containing a valid ThingSpace Bearer token.
VZ-M2M-Token required	string	A valid session token returned by a Connectivity Management API POST /session/login request using the account's UWS credentials.

Content-Type required	string	Must be `application/json`.

Request Body

The device identifier and fields to match in the search.

Property Name	Data Type	Description
accountidentifier required	JSON object	The ID of the authenticating billing account, in the format `{"billingaccountid":"1234567890-12345"}`.
resourceidentifier required	JSON object	A device ID type and value that identifies the device to search. Can be any unique device identifier, such as ICCID, ID, or IMEI.
$selection optional	JSON object	A comma-separated list of properties and comparator values to match against event data. See Working with Query Filters for more information. If the request does not include `$selection`, the response will include all stored event data.
$limitnumber optional	integer	The maximum number of events to include in the response. If the request matches more than this number of events, the response will contain an `X-Next` value in the header that can be used as the `page` value in the next request to retrieve the next page of events.
$page optional	string	Unique string from the previous response to retrieve either the next page (`X-Next` value) or the previous page (`X-Prev` value) of events.

Example Request Body

Find the two most recent configuration events

```
curl POST https://thingspace.verizon.com/api/cc/v1/devices/fields/actions/history/search
\
  -H 'Authorization: Bearer 021833fbee3e118019f67777e028067f' \
  -H 'VZ-M2M-Token: f1ffff45-ed4c-4db0-a98c-a371afe23b6a' \
  -H 'Content-Type: application/json' \
  -d '{
    "accountidentifier": {"billingaccountid":"1223334444-00001"},
    "resourceidentifier": {"imei":864508030084997},
```

```
    "$selection": {"kind":"ts.event.configuration"},
    "$limitnumber": 2
  }'
```

Success Responses

Status 200

A success response includes an array of all matching events. Each event includes the full event resource definition.

Example Success Response


```
[
  {
    "action": "set",
    "createdon": "2019-02-21T02:05:25.270417481Z",
    "description": "[200]-OK",
    "deviceid": "8461f530-2e31-6e87-e357-6c38251d4d01",
    "fields": {
      "configuration": {
        "frequency": "Medium",
        "location_mode": "gps"
      }
    },
    "id": "e521b8ae-440d-6cc1-f687-7c80e085ff29",
    "kind": "ts.event.configuration",
    "lastupdated": "2019-02-21T02:05:25.394230017Z",
    "name": "SetConfigurationReq",
    "state": "update",
    "tagids": [
      "4d110e4f-7a7c-6b26-eaac-31cc34c6e1d4",
      "cd81ed16-18ae-6c7d-eaba-2883b0395387"
    ],
    "transactionid": "c7a0a8cf-6856-4733-93ea-69913650e4ca",
    "version": "1.0",
    "versionid": "270e4820-357d-11e9-9d6c-02420a470c06"
  },
  {
    "action": "set",
    "createdon": "2019-02-21T01:58:45.163Z",
    "deviceid": "8461f530-2e31-6e87-e357-6c38251d4d01",
    "fields": {
      "configuration": {
        "frequency": "Medium",
        "location_mode": "gps"
      }
    },
    "id": "25819e96-c4f2-6545-fb68-0e1d9662359f",
    "kind": "ts.event.configuration",
    "lastupdated": "2019-02-21T01:58:45.163Z",
    "state": "pending",
```

```
    "tagids": [
      "4d110e4f-7a7c-6b26-eaac-31cc34c6e1d4",
      "cd81ed16-18ae-6c7d-eaba-2883b0395387"
    ],
    "transactionid": "c7a0a8cf-6856-4733-93ea-69913650e4ca",
    "version": "1.0",
    "versionid": "387fe2fe-357c-11e9-ae40-02420a49140a"
  }
]
```


Failure Responses

Status 4xx

All error messages are returned in this format:


```
{
  "error": "error code string",
  "error_description": "error message string",
  "cause": "further explanation"
}
```


Retrieve sensor history:

POST /devices/fields/{fieldname}/actions/history

Returns the readings of a specified sensor, with the most recent reading first. Sensor readings are stored as events; this request an array of events.

Request Components

HTTP Request

`POST https://thingspace.verizon.com/api/cc/v1/devices/fields/{fieldname}/actions/history`

Resource Path and Query Parameters

You must set the field (sensor) that you want to retrieve in the resource path.

Property Name	Data Type	Description
fieldname required	string	The name of the sensor.

Header Parameters

The request header must contain a current ThingSpace authorization token and a current VZ-M2M session token, and must set the content-type to JSON.

Property Name	Data Type	Description
Authorization required	string	HTTP Authorization request header containing a valid ThingSpace Bearer token.
VZ-M2M-Token required	string	A valid session token returned by a Connectivity Management API POST /session/login request using the account's UWS credentials.
Content-Type required	string	Must be `application/json`.

Request Body

The device identifier and the number of results that you want to retrieve.

Property Name	Data Type	Description
accountidentifier required	JSON object	The ID of the authenticating billing account, in the format `{"billingaccountid":"1234567890-12345"}`.
resourceidentifier required	JSON object	A device ID type and value that identifies the device to search. Can be any unique device identifier, such as ICCID, ID, or IMEI.

$limitnumber optional	integer	The maximum number of events to include in the response. If the request matches more than this number of events, the response will contain an `X-Next` value in the header that can be used as the `page` value in the next request to retrieve the next page of events.

Example Request Body

Retrieve the two most recent temperature values

```
curl POST
https://thingspace.verizon.com/api/cc/v1/devices/fields/temperature/actions/history \
  -H 'Authorization: Bearer 021833fbee3e118019f67777e028067f' \
  -H 'VZ-M2M-Token: f1ffff45-ed4c-4db0-a98c-a371afe23b6a' \
  -H 'Content-Type: application/json' \
  -d '{
  "accountidentifier": {"billingaccountid":"1223334444-00001"},
  "resourceidentifier": {"imei":864508030084997},
  "$limitnumber": 2
  }'
```

Success Responses

Status 200

A success response includes an array of all matching events. Each event includes the full event resource definition.

Example Success Response

```
[
  {
    "action": "update",
    "createdon": "2019-02-22T04:05:26Z",
    "deviceid": "8461f530-2e31-6e87-e357-6c38251d4d01",
    "fields": {
      "temperature": "18.4"
    },
    "id": "4521b7a7-7de1-6e68-f020-1345ef3b764a",
    "kind": "ts.event",
    "lastupdated": "2019-02-22T04:05:49.786Z",
    "state": "update",
    "tagids": [
      "4d110e4f-7a7c-6b26-eaac-31cc34c6e1d4",
      "cd81ed16-18ae-6c7d-eaba-2883b0395387"
    ],
    "transactionid": "864508030084997-OnBoard-c05f946c-3f5c-4527-b4d0-5aca256fc3dd",
    "version": "1.0",
    "versionid": "238addb9-3657-11e9-8848-02420a951f13"
  },
```

```
  {
    "action": "update",
    "createdon": "2019-02-22T03:05:26Z",
    "deviceid": "8461f530-2e31-6e87-e357-6c38251d4d01",
    "fields": {
      "temperature": "19.0"
    },
    "id": "05b1ea7b-4bf2-6af6-ea8b-414595f2c3e9",
    "kind": "ts.event",
    "lastupdated": "2019-02-22T03:05:48.483Z",
    "state": "update",
    "tagids": [
      "4d110e4f-7a7c-6b26-eaac-31cc34c6e1d4",
      "cd81ed16-18ae-6c7d-eaba-2883b0395387"
    ],
    "transactionid": "864508030084997-OnBoard-5f71f47d-4464-4f69-a3ee-5c243f0ef5b8",
    "version": "1.0",
    "versionid": "c0ffa4b5-364e-11e9-a3ee-02420a8c0d14"
  }
]
```


Failure Responses

Status 4xx

All error messages are returned in this format:

```
{
  "error": "error code string",
  "error_description": "error message string",
  "cause": "further explanation"
}
```

Search for Targets

POST /targets/actions/query

Search for targets by property values. Returns an array of all matching target resources.

Request Components

HTTP Request

`POST https://thingspace.verizon.com/api/cc/v1/targets/actions/query`

Resource Path and Query Parameters

None.

Header Parameters

The request header must contain a current ThingSpace authorization token and a current VZ-M2M session token, and must set the content-type to JSON.

Property Name	Data Type	Description
Authorization required	string	HTTP Authorization request header containing a valid ThingSpace Bearer token.
VZ-M2M-Token required	string	A valid session token returned by a Connectivity Management API POST /session/login request using the account's UWS credentials.
Content-Type required	string	Must be `application/json`.

Request Body

The request body specifies fields and values to match.

Property Name	Data Type	Description
accountidentifier required	JSON object	The ID of the authenticating billing account, in the format `{"billingaccountid":"1234567890-12345"}`.
$selection optional	JSON object	A comma-separated list of properties and comparator values to match against targets in the ThingSpace account. See Working with Query Filters for more information. If the request does not include `$selection`, the response will include all targets to which the requesting user has access.

Example Request

Find a target by its UUID

```
curl -X POST https://thingspace.verizon.com/api/cc/v1/targets/actions/query \
  -H 'Authorization: Bearer 225aec324bb619d10101ca905fd2479e' \
  -H 'VZ-M2M-Token: 85d73f80-3127-4d04-81ae-82497cc557dd' \
  -H 'Content-Type: application/json' \
  -d '{
    "accountidentifier": {"billingaccountid":"1223334444-00001"}
    "resourceidentifier": {"id": "dd1682d3-2d80-cefc-f3ee-25154800beff"}
  }'
```

Find all AWS targets

```
curl -X POST https://thingspace.verizon.com/api/cc/v1/targets/actions/query \
  -H 'Authorization: Bearer 225aec324bb619d10101ca905fd2479e' \
  -H 'VZ-M2M-Token: 85d73f80-3127-4d04-81ae-82497cc557dd' \
  -H 'Content-Type: application/json' \
  -d '{
    "accountidentifier": {"billingaccountid":"1223334444-00001"}
    "$selection": {"addressscheme": "streamawsiot"}
  }'
```

Success Responses

Status 200

A success response includes an array of all matching targets. Each target includes the full target resource definition.

Property Name	Data Type	Description
address	string	The endpoint for data streams.
addressscheme	string	The transport format.
createdon	string, date-time	The date the resource was created.
description	string	Description of the target.

Property Name	Data Type	Description
externalid	string	Security identification string.
id	string, UUID	ThingSpace unique ID for the target that was created.
kind	string	Identifies the resource kind. Targets are ts.target.
lastupdated	string, date-time	The date the resource was last updated.
name	string	Name of the target.
version	string	Version of the underlying schema resource.
versionid	string, UUID	The version of the resource.

Example Success Response


```
[
  {
    "address": "https://myhost.com:1825",
    "addressscheme": "streamrest",
    "createdon": "2018-12-22T03:59:18.563Z",
    "id": "cee10900-f54e-6eef-e455-bd7f15c4be32",
    "kind": "ts.target",
    "lastupdated": "2018-12-22T03:59:18.563Z",
    "name": "host:port target",
    "version": "1.0",
    "versionid": "f4be7c2b-059d-11e9-bec6-02420a4e1b0a"
  },
  {
    "address": "arn:aws:iam::252156542789:role/ThingSpace",
    "addressscheme": "streamawsiot",
    "createdon": "2019-01-24T19:06:43.577Z",
    "externalid": "lJZnih8BfqsosZrEEkfPuR3aGOk2i-
HIr6tXN275ioJF6bezIrQB9EbzpTRep8J7RmV7QH==",
    "id": "cea170cc-a58f-6531-fc4b-fae1ceb1a6c8",
    "kind": "ts.target",
```

```
    "lastupdated": "2019-01-24T19:32:31.841Z",
    "name": "AWS Target",
    "region": "us-east-1",
    "version": "1.0",
    "versionid": "caf85ff7-200e-11e9-a85b-02420a621e0a"
  }
]
```


Failure Responses

Status 4xx
All error messages are returned in this format:


```
{
  "error": "error code string",
  "error_description": "error message string",
  "cause": "further explanation"
}
```


Error codes and messages are listed on the Error Messages page, along with explanations and suggestions for corrective actions.

Delete a target:

POST /targets/actions/delete

Remove a target from a ThingSpace account.

Request Components

HTTP Request

`POST https://thingspace.verizon.com/api/cc/v1/targets/actions/delete`

Resource Path and Query Parameters

None.

Header Parameters

The request header must contain a current ThingSpace authorization token and a current VZ-M2M session token, and must set the content-type to JSON.

Property Name	Data Type	Description
Authorization required	string	HTTP Authorization request header containing a valid ThingSpace Bearer token.
VZ-M2M-Token required	string	A valid session token returned by a Connectivity Management API POST /session/login request using the account's UWS credentials.
Content-Type required	string	Must be `application/json`.

Request Body

The request body identifies the target to delete.

Property Name	Data Type	Description
accountidentifier required	JSON object	The ID of the authenticating billing account, in the format `{"billingaccountid":"1234567890-12345"}`.
resourceidentifier required	JSON object	The ID of the target to delete, in the format `{"id": "dd1682d3-2d80-cefc-f3ee-25154800beff"}`.

Example Request

```
curl -X POST \
  https://thingspace.verizon.com/api/cc/v1/targets/actions/delete \
  -H 'Authorization: Bearer 225aec324bb619d10101ca905fd2479e' \
  -H 'Cache-Control: no-cache' \
  -H 'Content-Type: application/json' \
  -H 'VZ-M2M-Token: 85d73f80-3127-4d04-81ae-82497cc557dd' \
  -d '{
    "accountidentifier": {"billingaccountid":"1223334444-00001"},
    "resourceidentifier": {"id": "aef1d746-16b1-689b-ee9f-45c34960a427"}
```

```
}'
```

Success Responses

Status 204

Target deleted successfully. The response has no body.

Failure Responses

Status 4xx

All error messages are returned in this format:

```
{
  "error": "error code string",
  "error_description": "error message string",
  "cause": "further explanation"
}
```

Search for Subscriptions

POST /subscriptions/actions/query

Search for subscriptions by property values. Returns an array of all matching subscription resources.

Request Components

HTTP Request

```
POST https://thingspace.verizon.com/api/cc/v1/subscriptions/actions/query
```

Resource Path and Query Parameters

None.

Header Parameters

The request header must contain a current ThingSpace authorization token and a current VZ-M2M session token, and must set the content-type to JSON.

Property Name	Data Type	Description
Authorization required	string	HTTP Authorization request header containing a valid ThingSpace Bearer token.
VZ-M2M-Token required	string	A valid session token returned by a Connectivity Management API POST /session/login request using the account's UWS credentials.
Content-Type required	string	Must be `application/json`.

Request Body

The request body specifies fields and values to match.

Property Name	Data Type	Description
accountidentifier required	JSON object	The ID of the authenticating billing account, in the format `{"billingaccountid":"1234567890-12345"}`.
$selection optional	JSON object	A comma-separated list of properties and comparator values to match against subscriptions in the ThingSpace account. See Working with Query Filters for more information. If the request does not include `$selection`, the response will include all subscriptions to which the requesting user has access.

Example Request

Find a subscription by its UUID

```
curl -X POST https://thingspace.verizon.com/api/cc/v1/subscriptions/actions/query \
  -H 'Authorization: Bearer 506be090c32836bad23e1bcbbe829820' \
  -H 'VZ-M2M-Token: 87734bdb-9c00-4f9a-96f1-d2cc800624b8' \
  -H 'Content-Type: application/json' \
  -d '{
    "accountidentifier": {"billingaccountid":"1223334444-00001"},
    "resourceidentifier": {"id": "dd1682d3-2d80-cefc-f3ee-25154800beff"}
  }'
```

Find all subscription that have configuration failures

```
curl -X POST https://thingspace.verizon.com/api/cc/v1/subscriptions/actions/query \
  -H 'Authorization: Bearer 506be090c32836bad23e1bcbbe829820' \
  -H 'VZ-M2M-Token: 87734bdb-9c00-4f9a-96f1-d2cc800624b8' \
  -H 'Content-Type: application/json' \
  -d '{
    "accountidentifier": {"billingaccountid":"1223334444-00001"},
    "$selection": {"configurationfailures gt": 0 }
  }'
```

Success Responses

Status 200
A success response includes an array of all matching subscriptions. Each subscription includes the full subscription resource definition.

Property Name	Data Type	Description
configurationfailures	integer	The number of streaming failures due to faulty configuration.
createdon	string, date-time	The date the resource was created.
delegateid	string	Not currently used.
description	string	Description of the subscription.
disabled	boolean	Whether the subscription is currently sending data.
email	string	The address to which any error reports should be delivered.
filter	string	Filter for events.
groupid	string	ID of a tag resource for group-level streaming.
id	string, UUID	ThingSpace unique ID for the subscription that was created.
kind	string	Identifies the resource kind.
laststreamingstatus	string	Success or fail

Property Name	Data Type	Description
laststreamingtime	string, date-time	The date and time that the last stream send was attempted.
lastupdated	string, date-time	The date the resource was last updated.
name	string	Name of the subscription.
networkfailures	integer	The number of failures due to network problems.
streamfailures	integer	
streamkind	string, enum	The event type that will be sent in the data stream.
targetid	string	Target to be used for dispatching events.
targettype	string	
version	string, UUID	Version of the underlying schema resource.
versionid	string, UUID	The version of the resource.

Example Success Response


```
[
  {
    "configurationfailures": 0,
    "createdon": "2019-02-13T23:13:24.689Z",
    "delegateid": "00000000-0000-0000-0000-000000000000",
    "disabled": false,
    "email": "me@mycompany.com",
    "id": "98015aed-e984-62be-f049-1d895d2d1812",
    "kind": "ts.subscription",
    "laststreamingstatus": "success",
    "laststreamingtime": "2019-02-20T18:20:45.865Z",
    "lastupdated": "2019-02-13T23:13:24.689Z",
    "networkfailures": 0,
    "streamfailures": 0,
    "streamkind": "ts.event",
```

```
    "targetid": "4e112cb3-da1d-6ece-f2c6-bb8700b20b09",
    "targettype": "Amazon Web Services",
    "version": "1.0",
    "versionid": "f68b8862-2fe4-11e9-85fd-02420a4c170d"
  }
]
```

Failure Responses

Status 4xx
All error messages are returned in this format:

```
{
  "error": "error code string",
  "error_description": "error message string",
  "cause": "further explanation"
}
```

Error codes and messages are listed on the Error Messages page, along with explanations and suggestions for corrective actions.

Delete a subscription:

POST /subscriptions/actions/delete

Remove a subscription from a ThingSpace account.

Request Components

HTTP Request

`POST https://thingspace.verizon.com/api/cc/v1/subscriptions/actions/delete`

Resource Path and Query Parameters

None.

Header Parameters

The request header must contain a current ThingSpace authorization token and a current VZ-M2M session token, and must set the content-type to JSON.

Property Name	Data Type	Description
Authorization required	string	HTTP Authorization request header containing a valid ThingSpace Bearer token.
VZ-M2M-Token required	string	A valid session token returned by a Connectivity Management API POST /session/login request using the account's UWS credentials.
Content-Type required	string	Must be `application/json`.

Request Body

The request body identifies the subscription to delete.

Property Name	Data Type	Description
accountidentifier required	JSON object	The ID of the authenticating billing account, in the format `{"billingaccountid":"1234567890-12345"}`.
resourceidentifier required	JSON object	The ID of the subscription to delete, in the format `{"id": "dd1682d3-2d80-cefc-f3ee-25154800beff"}`.

Example Request

```
curl -X POST https://thingspace.verizon.com/api/cc/v1/subscriptions/actions/delete \
  -H 'Authorization: Bearer 18261b8341d07fc32c760c2a5acaad27' \
  -H 'VZ-M2M-Token: 885b3ba5-b149-4fe8-91c5-51efb1b7462a' \
  -H 'Content-Type: application/json' \
  -d '{
    "accountidentifier": {"billingaccountid":"1223334444-00001"},
    "resourceidentifier": {"id": "f8b112df-739c-6236-f059-106c67bafd99"}
}'
```

Success Responses

Status 204

Subscription deleted successfully. The response has no body.

Failure Responses

Status 4xx

All error messages are returned in this format:

```
{
  "error": "error code string",
  "error_description": "error message string",
  "cause": "further explanation"
}
```

Completion/Returns

Upon completion of the pilot, please return the device to the address listed below. Data will continue to be streamed to the customer's backend until Verizon is able to disable and disconnect the devices.

All device returns should be sent to the attention of :

Rob Arlic, Verizon
15505 Sand Canyon Avenue
Building C
Irvine, CA 92618.

Also, e-mail rob.arlic@verizon.com with Airway Bill Number and date of dispatch.