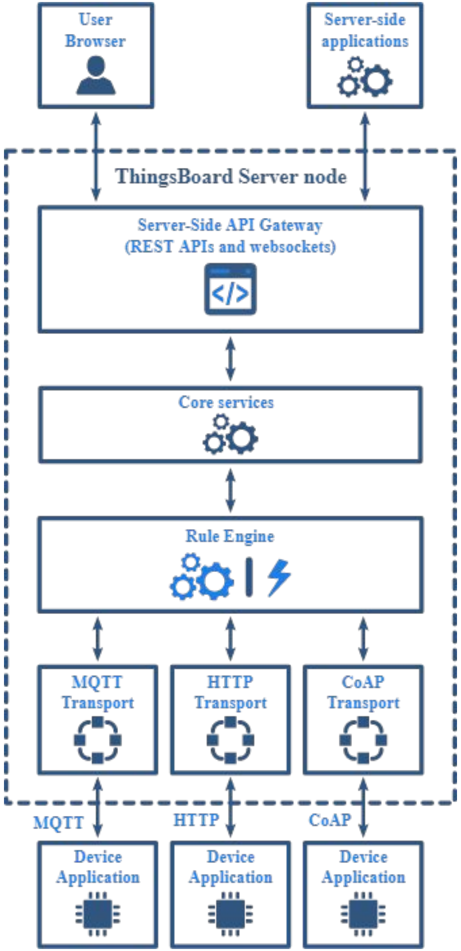
Introduction to IoT Platform: ThingsBoard

ThingsBoard High-Level Architecture



Entities and Relations

ThingsBoard provides many entities to develop and manage IoT applications.

1. Entity Types
    a. Tenants, Customers, Users
    b. Asset
    c. Device
    d. Alarms, Dashboard, Rule Node, Rule Chains
2. Relations
    a. Contains, Manages
    b. Custom
3. Attributes
4. Telemetry

Telemetry and RPC with ThingsBoard

Telemetry with MQTT

1. Supports **QoS 0** and **QoS 1**
2. **Predefined MQTT topics**
3. **Authentication methods**
   a. **Access token**
   b. X.509 Certificates

Publish data with MQTT

MQTT Topic: v1/devices/me/telemetry

Data should be sent as JSON

```
{"stringKey":"value1", "booleanKey":true, "doubleKey":42.0, "longKey":73}
```

Or

```
[{"stringKey":"value1", "doubleKey":42.0, "longKey":73}, {"stringKey":"value2", "doubleKey":45.5,
"longKey":53},]
```

To specify timestamp, the message format changes to

```
{"ts":1451649600512, "values":{"key1":"value1", "key2":"value2"}}
```

Publish data with MQTT

MQTT Topic: v1/devices/me/telemetry

Authentication requires $ACCESS_TOKEN to be sent in **username** field

1. In mosquitto_pub, use: “-u” “$ACCESS_TOKEN”
2. In Mongoose OS, specify “user” in “mqtt” part of the configuration

How to receive commands to device? (MQTT - RPC)

ThingsBoard provides RPC API over MQTT

Devices will SUBSCRIBE to commands on topic v1/devices/me/rpc/request/+

Responses (if required) should be PUBLISHed at topic
v1/devices/me/rpc/response/$request_id

How to send commands to device? (MQTT - RPC)

Data should be sent as a **JSON string** with two necessary keys
method: String, **params**: JSON Object

```
{
        "method": "setGpio",
        "params": {
                "pin": "23",
                "value": 1
        }
}
```

How to send commands to device? (MQTT - RPC)

To send commands to the device, you make an **HTTP POST** request with **JSON string** in body using data format shown previously, at given endpoint

http(s)://host:port/api/plugins/rpc/{callType}/{deviceId}

callType: oneway OR twoway
deviceId: can be found in ThingsBoard device UI

Specify **X-Authorization: Bearer $JWT_TOKEN** in HTTP header of the user authorized to use the device.

Off-topic: Obtaining JWT Token for a user

Make a HTTP POST request at given endpoint

http(s)://host:port/api/auth/login

In body, send a JSON string which contains username and password

```
{
    “username”: “tenant@thingsboard.org”,
    “password”: “tenant”
}
```

Telemetry with HTTP

1. HTTP version 1.1
2. Device Authentication using $ACCESS_TOKEN

Publish data with HTTP

Send a HTTP POST request with JSON string in body at given endpoint

http(s)://host:port/api/v1/$ACCESS_TOKEN/telemetry

Data should be sent as JSON

{"stringKey":"value1", "booleanKey":true, "doubleKey":42.0, "longKey":73}

Or

[{"stringKey":"value1", "doubleKey":42.0, "longKey":73}, {"stringKey":"value2", "doubleKey":45.5, "longKey":53},]

To specify timestamp, the message format changes to:

{"ts":1451649600512, "values":{"key1":"value1", "key2":"value2"}}

How to send commands to device? (HTTP - RPC)

ThingsBoard provides RPC API over HTTP

Devices can listen to RPC commands by polling at endpoint
http(s)://host:port/api/v1/$ACCESS_TOKEN/rpc?timeout={}

Timeout is optional and can be used to listen for set amount of milliseconds.

A response should be delivered using HTTP POST request at endpoint
http(s)://host:port/api/v1/$ACCESS_TOKEN/rpc/{$id}
where **id** is the request ID found in body of above GET request

How to send commands to device? (HTTP - RPC)

Same as with MQTT

Data should be sent as a **JSON string** with two necessary keys

Method: String, **Params**: JSON Object

```
    {
        "method": "setGpio",
        "params": {
            "pin": "23",
            "value": 1
        }
    }
```

How to send commands to device? (HTTP - RPC)

To send commands to the device, you make an **HTTP POST** request with **JSON string** in body using data format shown previously at given endpoint

http(s)://host:port/api/plugins/rpc/{callType}/{deviceId}

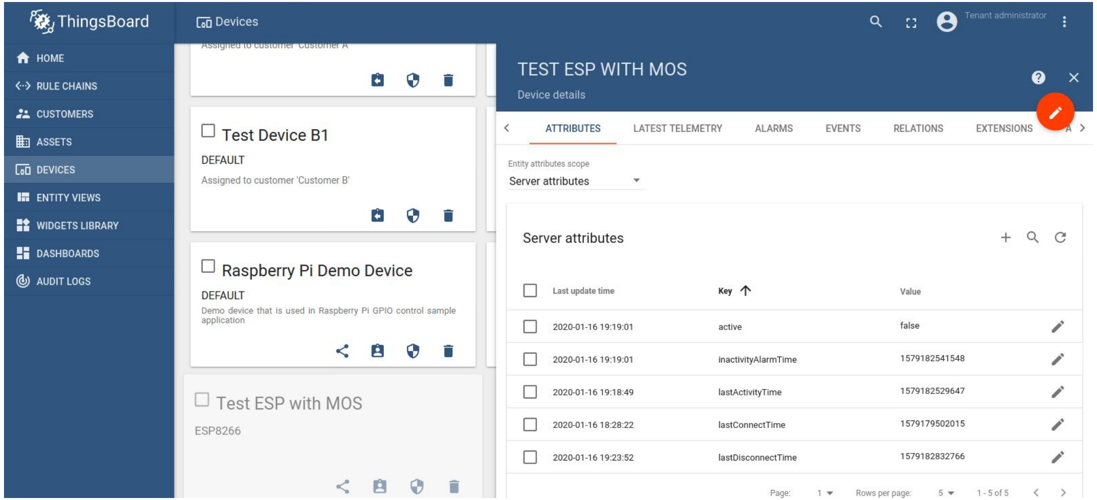
callType: oneway OR twoway (waits for a response)
deviceId: can be found in ThingsBoard device panel

Specify **X-Authorization: Bearer $JWT_TOKEN** in HTTP header of the user authorized to use the device.

Device State Service

Responsible for maintaining

1. Activity status
2. Last Connect Time
3. Last Disconnect Time
4. Inactivity Alarm Time
5. Last Activity Time



Assigned to customer 'Customer A'

□ Test Device B1

DEFAULT

Assigned to customer 'Customer B'

□ Raspberry Pi Demo Device

DEFAULT

Demo device that is used in Raspberry Pi GPIO control sample application

□ Test ESP with MOS

ESP8266

TEST ESP WITH MOS

Device details

Entity attributes scope: Server attributes

Server attributes

□	Last update time	Key ↑	Value
□	2020-01-16 19:19:01	active	false
□	2020-01-16 19:19:01	inactivityAlarmTime	1579182541548
□	2020-01-16 19:18:49	lastActivityTime	1579182529647
□	2020-01-16 18:28:22	lastConnectTime	1579179502015
□	2020-01-16 19:23:52	lastDisconnectTime	1579182832766

Page: 1 Rows per page: 5 1 - 5 of 5

Transferring Attributes

Why do we need attributes?

Telemetry reports sensor and actuator states (device state).

Where should we store semi-static data like:

1. Firmware-version
2. Location of the device
3. Other configuration parameters?

Answer: **Attributes**

Transferring Attributes

1. Device-side or Client-side
2. Server-side
3. Shared

Depending upon the application and nature of information, we use choose a particular type of attribute to persist the information.

Publishing Data

Authentication schemes and message format is same as with telemetry

Client-side attributes

1. Over MQTT: **v1/devices/me/attributes**
2. Over HTTP: **http(s)://host:port/api/v1/$ACCESS_TOKEN/attributes**

Publishing Data

Authentication schemes and message format is same as with telemetry (using $JWT_TOKEN)

Server-side attributes

1. Over HTTP:
   http(s)://host:port/api/plugins/telemetry/$DEVICE_ID/SERVER_SCOPE

Publishing Data

Authentication schemes and message format is same as with telemetry (using $JWT_TOKEN)

Shared attributes

1. Over HTTP:
   http(s)://host:port/api/plugins/telemetry/$DEVICE_ID/SHARED_SCOPE

Client-side RPC (for rules)

References

1. https://thingsboard.io/docs/