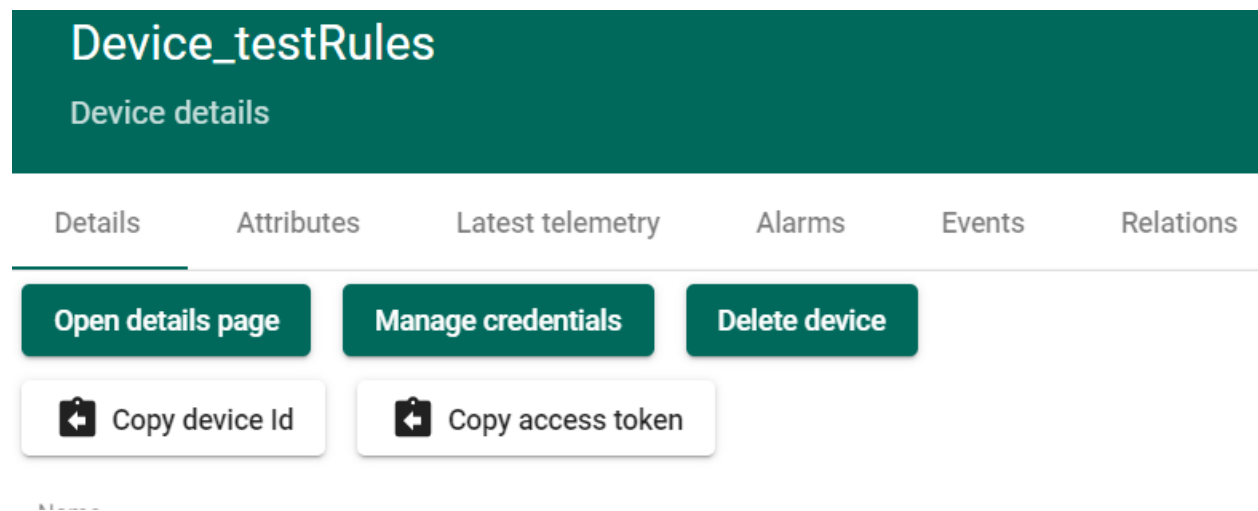


## D4 การทดลองที่ 1

- ทำการทดสอบตามเอกสาร Lab-401

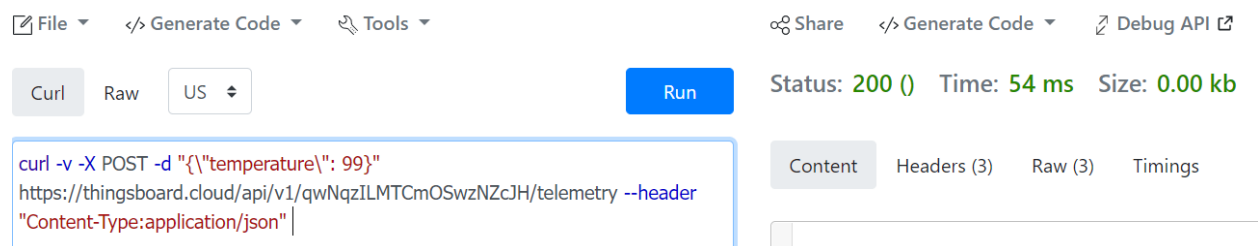
ทดสอบการใช้งาน Rule Chain เพื่อส่งการแจ้งเตือนไปยัง LINE

1.ที่ Thingsboard ทำการสร้าง Device เพื่อรับข้อมูล



2.นำ Token ที่ได้มาลองส่งข้อมูล Test curl on line from <https://reqbin.com/curl>

```
curl -v -X POST -d "{\"temperature\": 29.30, \"Humidity\": 58.11}"  
https://thingsboard.cloud/api/v1/qwNqzILMTCmOSwzNZcJH/telemetry --header "Content-Type:application/json"
```



3.สร้าง rule chain ใหม่แล้วกำหนดให้เป็น Root

ruleChain\_tastRules

Rule chain details

?

×

Details

Attributes

Latest telemetry

Alarms

Events

Relations

Audit Logs

Open rule chain

Export rule chain

Make rule chain root

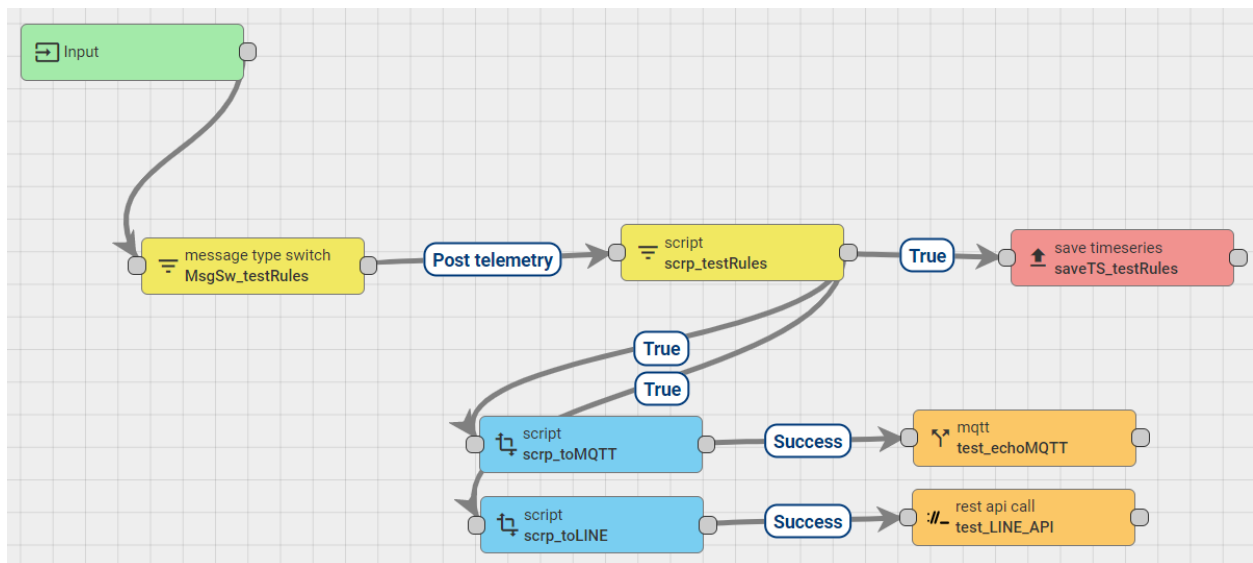
Delete rule chain

Copy rule chain Id

Name

ruleChain\_tastRules

4.สร้าง rule chain และกำหนดค่า ดังรูป



Scrp_testRules
<pre>return typeof msg.temperature === 'undefined'    (msg.temperature &gt;= -40 &amp;&amp; msg.temperature &lt;= 80);</pre>
Scrp_toMQTT
<pre>var newMsg = [] newMsg = "Temp=" + msg.temperature + ", Hudmid=" + msg.humidity return {msg: newMsg, metadata: metadata, msgType: msgType};</pre>
Scrp_toLINE
<pre>var newMsg = "Overheat, Tempperature = " + msg.temperature + "°C"; var newmetadata = { message: newMsg };</pre>

```
var msgType = "Debug Mode";  
return {msg: newMsg, metadata: newmetadata, msgType: msgType};
```

### Test\_echoMQTT

## test\_echoMQTT

External - mqtt

?

×

✓

×

Details

Events

Help

Name \*

test\_echoMQTT

☐ Debug mode

Topic pattern \*

monitorTB

Hint: use \${metadataKey} for value from metadata, \${messageKey} for value from message body

Host \*

test.mosquitto.org

Port \*

1883

Connection timeout (sec) \*

10

Client ID

Hint: Optional. Leave empty for auto-generated Client ID. Be careful when specifying the Client ID. Majority of the MQTT brokers will not allow multiple connections with the same Client ID. To connect to such brokers, your mqtt Client ID must be unique. When platform is running in a micro-services mode, the copy of rule node is launched in each micro-service. This will automatically lead to multiple mqtt clients with the same ID and may cause failures of the rule node. To avoid such failures enable "Add Service ID as suffix to Client ID" option below.

☐ Add Service ID as suffix to Client ID

Hint: Optional. Applied when "Client ID" specified explicitly. If selected then Service ID will be added to Client ID as a suffix. Helps to avoid failures when platform is running in a micro-services mode.

☒ Clean session

☐ Enable SSL

### Test\_LINE\_API

test\_LINE\_API

External - rest api call

Details

Events

Help

Name \*

test\_LINE\_API

Endpoint URL pattern \*

https://notify-api.line.me/api/notify?message=\${message}

Hint: use \${metadataKey} for value from metadata, \${messageKey} for value from

Request method

POST

Headers

Use \${metadataKey} for value from metadata, \${messageKey} for value from message body in header/value fields

Header	Value	
Content-Type	application/x-www-form-urlencoded	×
Authorization	Bearer <div>LINE Token</div>	×

5.ลองทดสอบส่งข้อมูลอีกครั้งแล้วสังเกตข้อมูลจาก MQTTLens และ LINE

<

Connection: asdf

Subscribe

monitorTB

0 - at most once

SUBSCRIBE

Publish

monitorTB

0 - at most once

☐ Retained

PUBLISH

Message

Subscriptions

Topic: "monitorTB" Showing the last 5 messages — +

Messages: 0/16

# Time Topic QoS

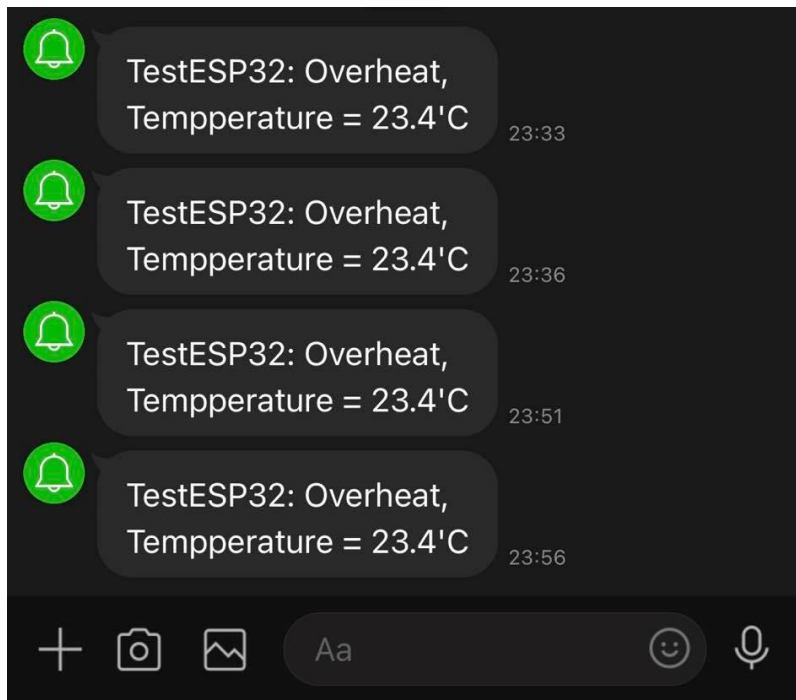
11 11:49:38 monitorTB 0

Message: "Temp=35, Hudmid=76.5"

# Time Topic QoS

12 11:50:07 monitorTB 0

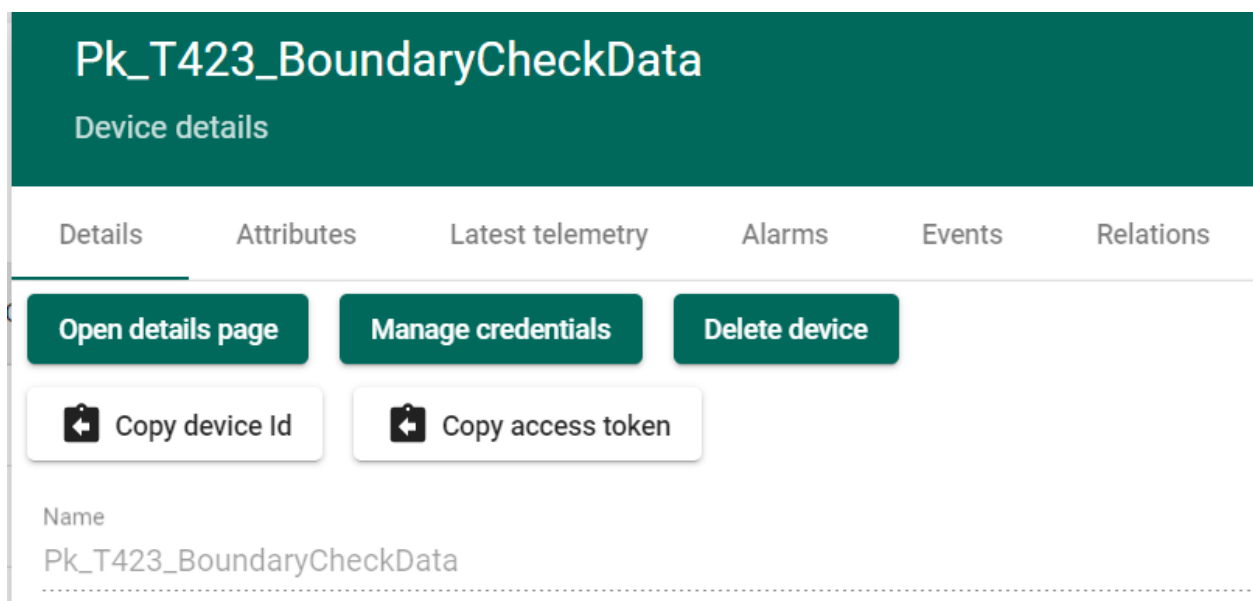
Message: "Temp=35, Hudmid=76.5"



#### D4 การทดลองที่ 2

ทำการทดสอบตามเอกสาร Lab-402 กำหนดเงื่อนไขในช่วงที่ยอมรับ คือ temperature = [-5,15] และ humidity = [40 – 60]%

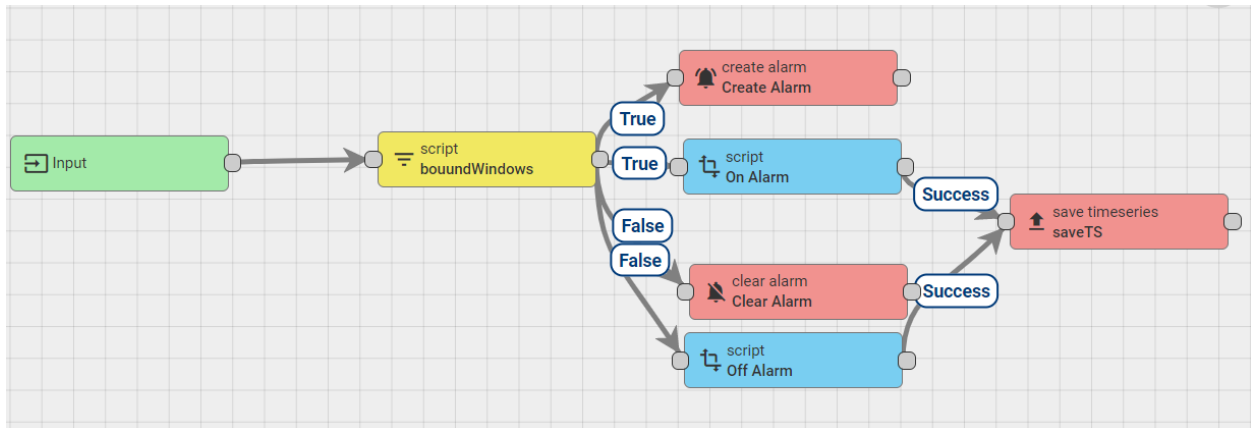
1.ที่ Thingsboard ทำการสร้าง Device เพื่อรับข้อมูล



2.นำ Token ที่ได้มาลองส่งข้อมูล Test curl on line from <https://reqbin.com/curl>

```
curl -v -X POST -d "{\"temperature\": 29.30, \"Humidity\": 58.11}"  
https://thingsboard.cloud/api/v1/WBhRjpio6BILcrCdf8yT/telemetry --header "Content-  
Type:application/json"
```

3.สร้าง rule chain ใหม่และกำหนดค่าดังนี้



Script: boundWindows

```
return (msg.temperature < -5 || msg.temperature > 15) || (msg.humidity < 40 ||  
msg.humidity > 60);
```

Script: On Alarm

```
msg.xAlarm = 1;  
return {msg: msg, metadata: metadata, msgType: msgType};
```

Script: Off Alarm

```
msg.xAlarm = 0;  
return {msg: msg, metadata: metadata, msgType: msgType};
```

## Create Alarm

Action - create alarm

Details Events Help

Name \*

Create Alarm

☐ Debug mode

☐ Use message alarm data

Alarm details builder

function Details(msg, metadata, msgType) {

Tidy ?

```
1 var details = {};  
2 if (metadata.prevAlarmDetails) {  
3   details = JSON.parse(metadata.prevAlarmDetails);  
4   //remove prevAlarmDetails from metadata  
5   delete metadata.prevAlarmDetails;  
6   //now metadata is the same as it comes IN this rule node  
7 }  
8  
9  
10 return details;
```

}

Test details function

Alarm type \*

Critical Temperature

## Create Alarm

Action - create alarm

Details Events Help

Test details function

Alarm type \*

Critical Temperature

Hint: use \${metadataKey} for value from metadata, \${messageKey} for value from message body

☐ Use alarm severity pattern

Alarm severity \*

Critical

☐ Propagate alarm to related entities

☒ Propagate alarm to entity owner (Customer or Tenant)

☒ Propagate alarm to entity owners hierarchy

☒ Propagate alarm to Tenant

Description

Clear Alarm

Action - clear alarm

?

×

Details

Events

Help

Name \*

Clear Alarm

☐ Debug mode

Alarm details builder

function Details(msg, metadata, msgType) {

Tidy

?

⌕

```

1 var details = {};
2 if (metadata.prevAlarmDetails) {
3     details = JSON.parse(metadata.prevAlarmDetails);
4     //remove prevAlarmDetails from metadata
5     delete metadata.prevAlarmDetails;
6     //now metadata is the same as it comes IN this rule node
7 }
8
9
10 return details;

```

Test details function

Alarm type \*

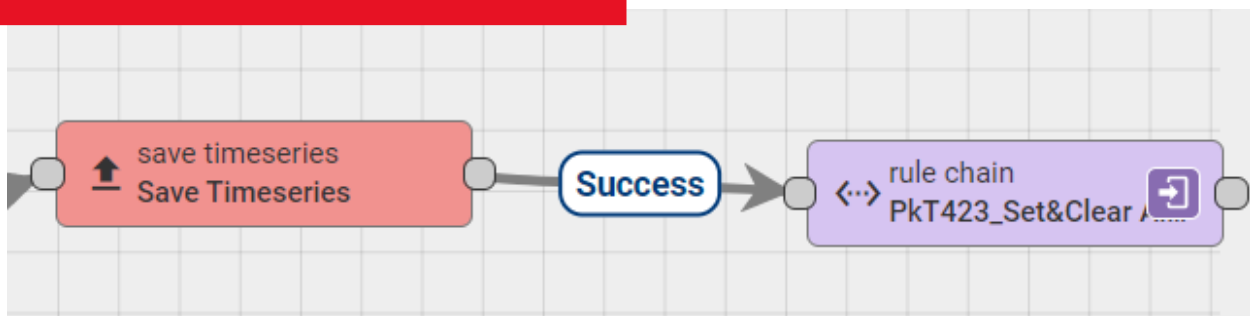
Critical Temperature

Hint: use \${metadataKey} for value from metadata, \${messageKey} for value from message body

#### 4.แก้ไขในส่วนของ Root Rule Chain

##### Rule chains

<input type="checkbox"/>	Created time ↓	Name
<input type="checkbox"/>	2022-05-10 00:00:48	PkT423_Set&Clear Alarm
<input type="checkbox"/>	2022-05-09 23:15:27	ruleChain_tastRules
<input type="checkbox"/>	2022-05-01 15:41:26	Root Rule Chain





## PkT423\_Set&Clear Alarm

Flow - rule chain

Details Events Help

**Open rule chain**

Name \*  
PkT423\_Set&Clear Alarm

Rule chain \*  
PkT423\_Set&Clear Alarm

5.สร้าง Dashboard จากนั้นเพิ่ม Entity Aliases

### Edit alias

Alias name \*  
pkT423\_Set&Clear Alarm

Resolve as multiple entities ☐

Filter type \*  
Single entity

Type \*  
Device

Device \*  
Pk\_T423\_BoundaryCheckData

Cancel Save

6.เพิ่ม Alarm Widget และตั้งค่า ดังรูป

### ← Alarm widgets: select widget

Type ↓	Severity
<input type="checkbox"/> Temperature	Major
<input type="checkbox"/> Temperature	Critical
<input type="checkbox"/> Low Humidity	Warning

#### Alarms table

Alarm widget

Displays alarms based on defined time window and other filters.

New Alarms table

Alarms table

Data

Settings

Advanced

Actions

Alarm type list

Any type

☐ Search propagated alarms

Alarm source

Entity alias \*

pkT423\_Set&Clear Alarm x

Entity

Filter

Created time: createdTime

Originator: originator

Type: type

Severity: severity

Status: status

## 7.เพิ่ม Horizon Bar และตั้งค่าตามรูป

Datasources

Maximum 1 datasource is allowed.

Type

Parameters

Entity alias \*

pkT423\_Set&Clear Alarm x

Entity

Filter

xAlarm: xAlarm

Maximum 1 timeseries/attribute is allowed.

## 7.เพิ่ม Digital Bar และตั้งค่าตามรูป

Entity alias \*

pkT423\_Set&Clear Alarm x

Entity

Filter

temperature: temperature

Maximum 1 timeseries/attribute is allowed.

## 8.เพิ่ม line chart และตั้งค่าตามรูป

Datasources

Type

Parameters

Entity alias \*

pkT423\_Set&Clear Alarm x

Entity

Filter

temperature: temperature

humidity: humidity

## 9.ทำการทดสอบการทำงาน

```
curl -v -X POST -d "{\"temperature\": 29.30, \"Humidity\": 58.11}"  
https://thingsboard.cloud/api/v1/WBhRjpio6BILcrCdf8yT/telemetry --header "Content-  
Type:application/json"
```

