login in phyclouds-

http://phyclouds.com:8080

id - students@thingsboard.org

password – students@sbcs

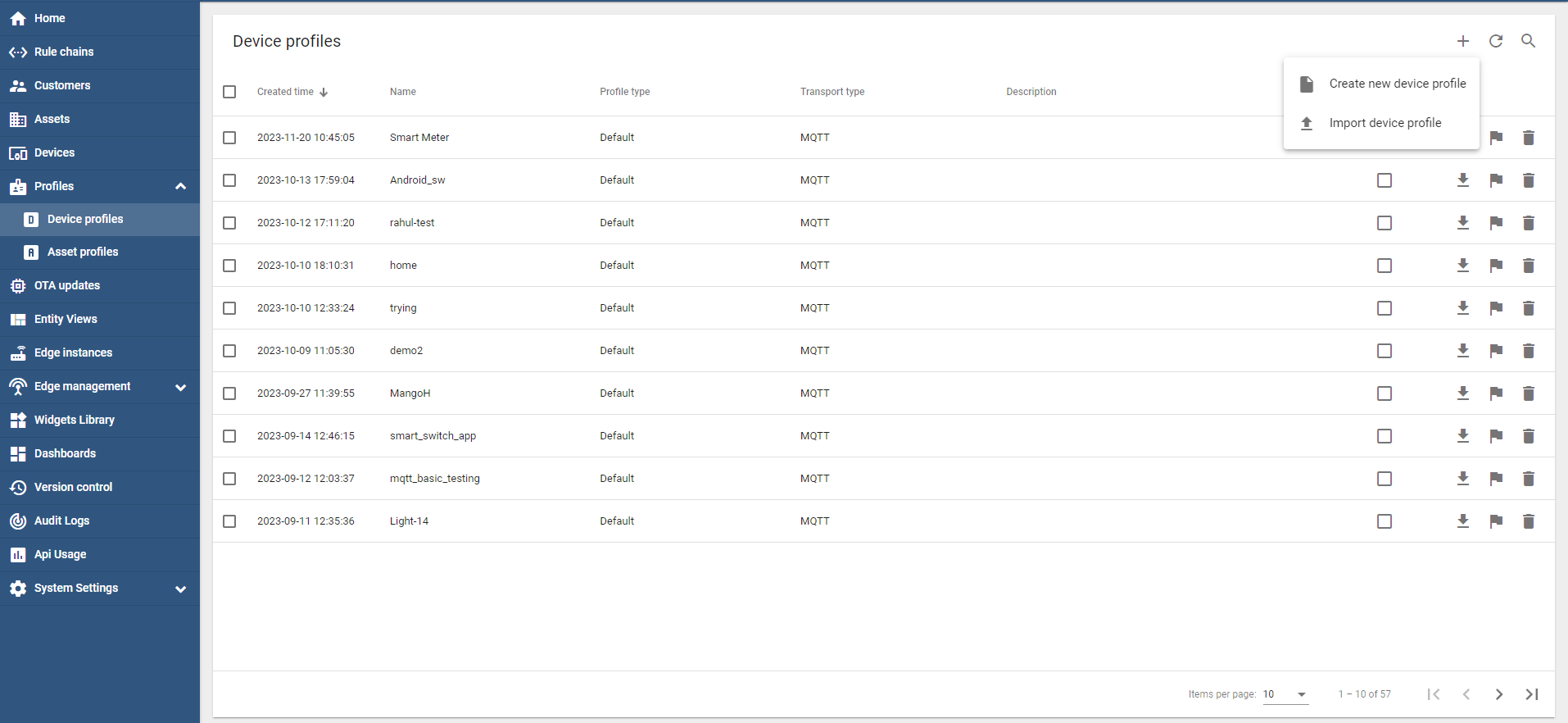
mqtt guide –

<https://thingsboard.io/docs/reference/mqtt-api/>

<https://thingsboard.io/docs/user-guide/ui/devices/>

how to create device (with MQTT communication)-

1. for device we have to create device profile only one profile is enough.



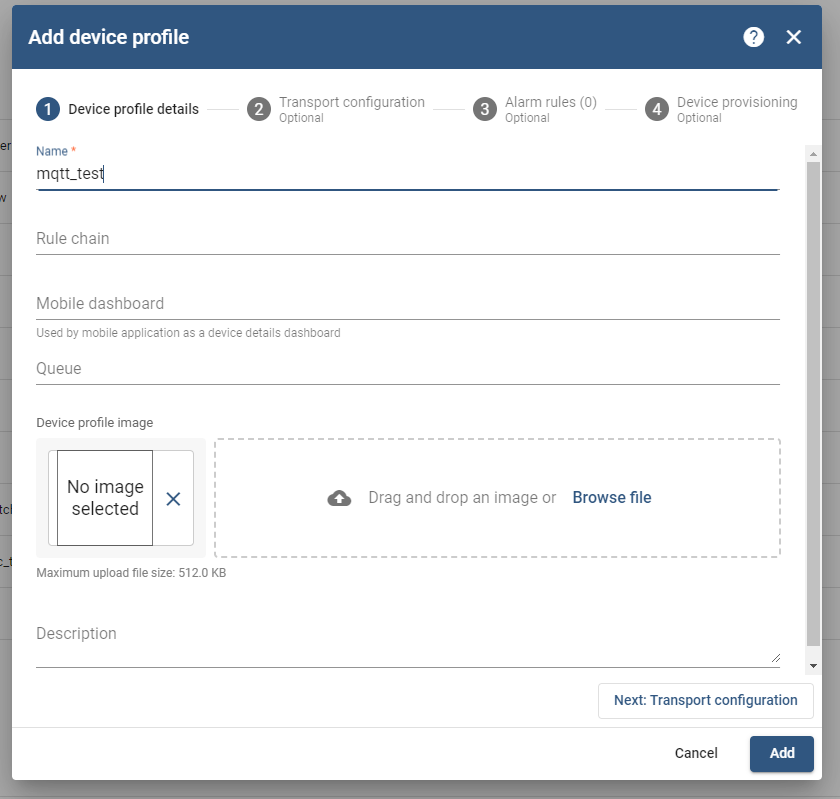
**22**

**1**

**3**

2. give all credential.

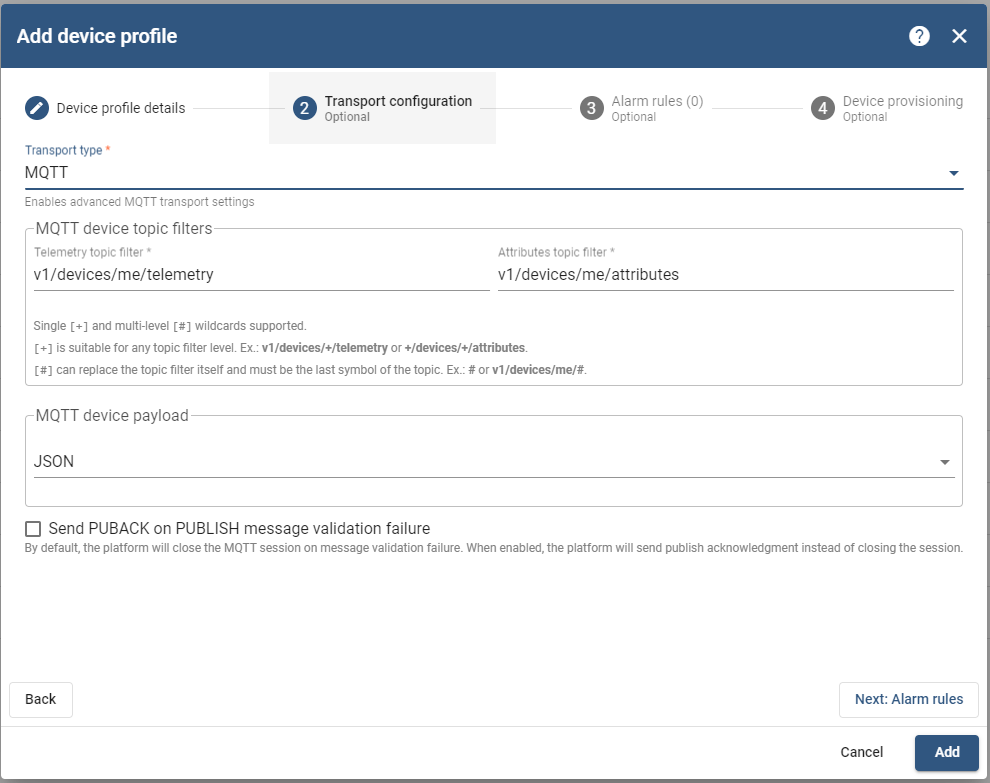
User give any name.



**2**

**1**

Next go to Transport Configuration -> Add

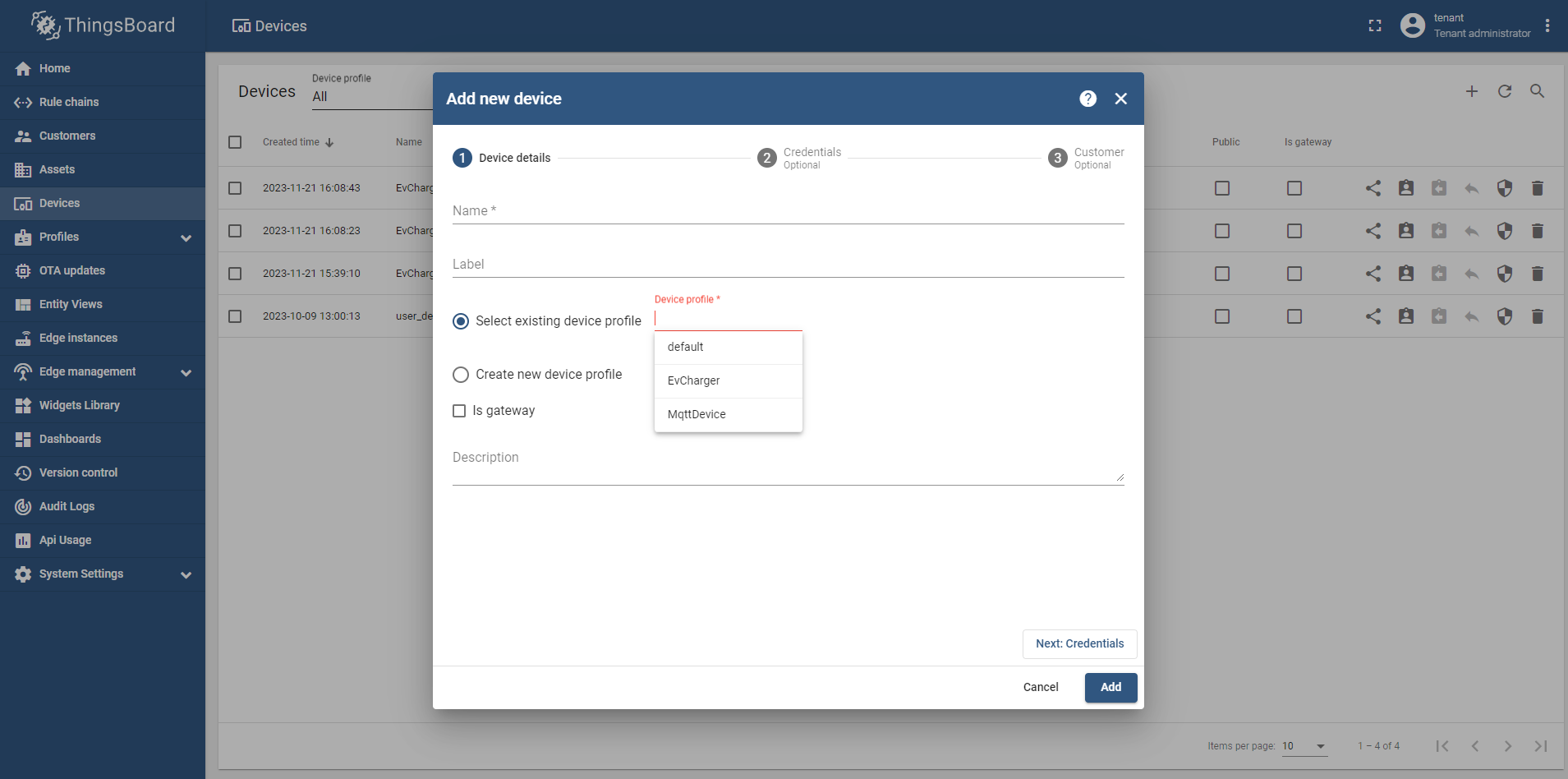


**1**

**2**

1. See devices on left side select then create device.

\*for communication access token we get from Devices.



**1**

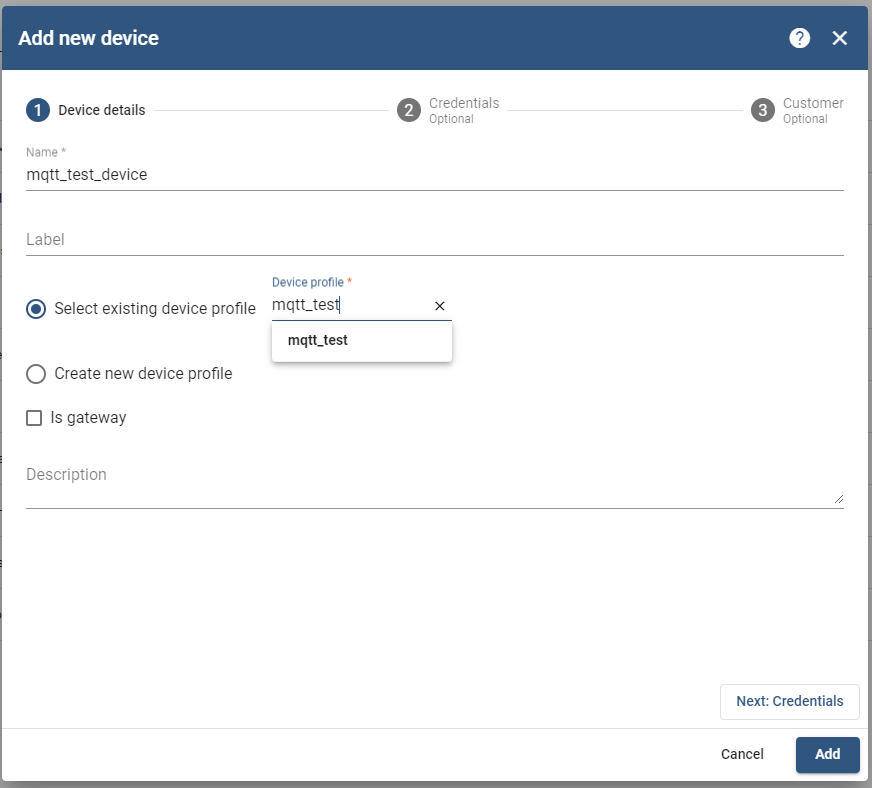
**2**

**3**

First remove default Device Profile then.

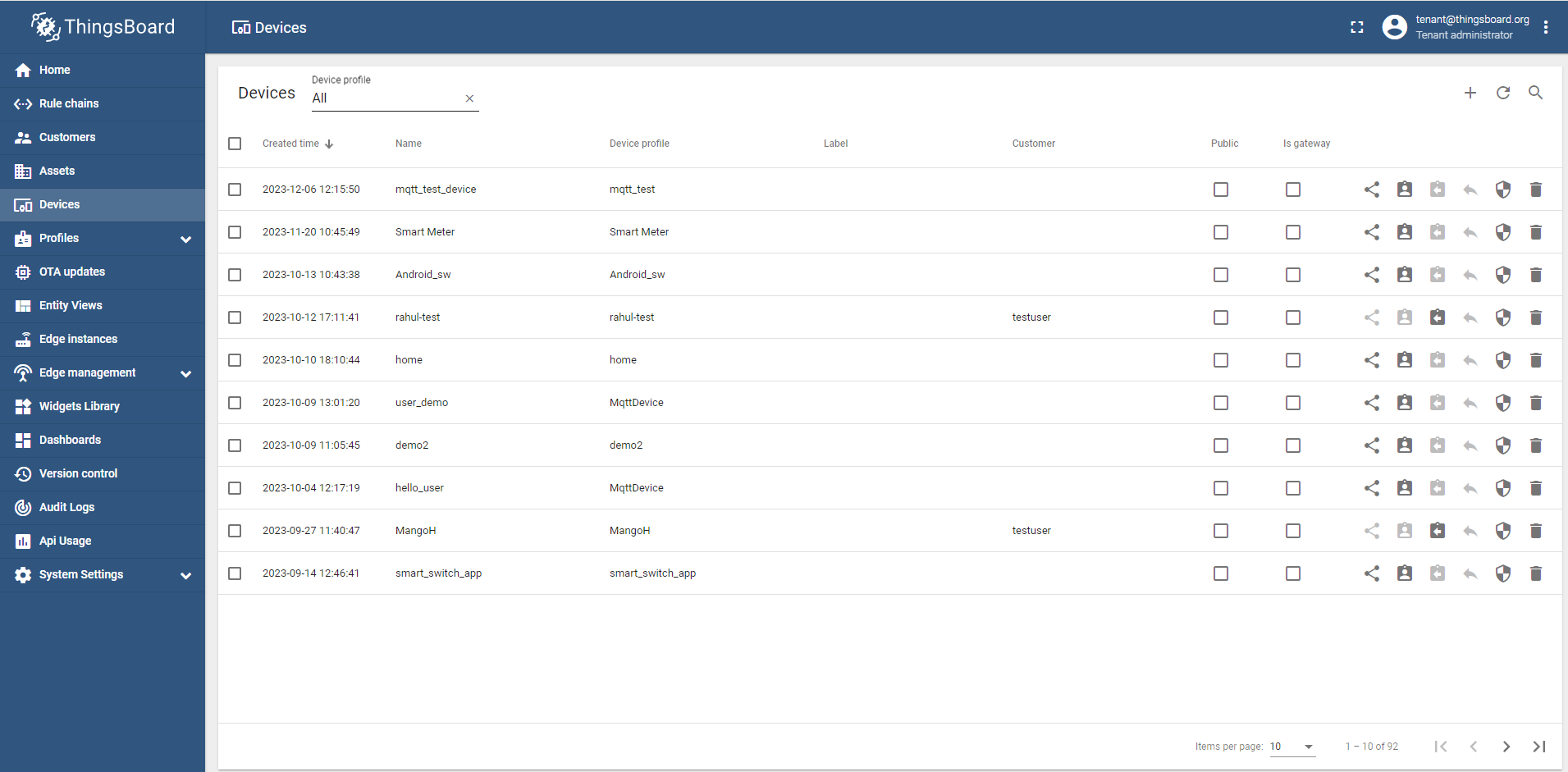
Give the Device profile what we created for mqtt communiocation.

Then click on Add button.



**1**

2. remember we have to give device profile in all devices what we created.

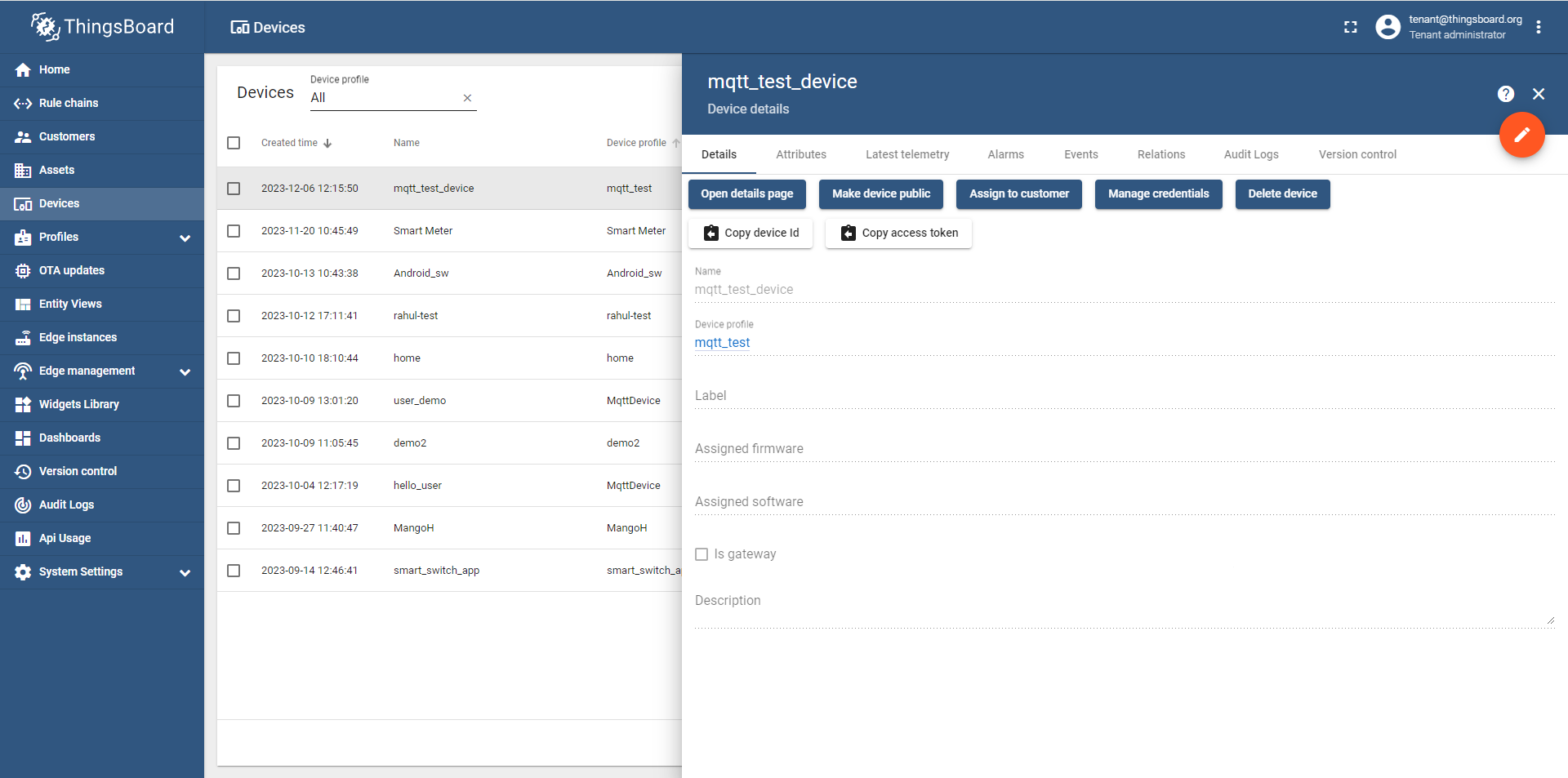


**1**

Here I am giving one python code -

import paho.mqtt.client as paho *# mqtt library*import time  
  
ACCESS\_TOKEN = 'zaPXMfOhBbByWel7XPf7' *# Token of your device*broker = "phyclouds.com" *# host name*port = 1884 *# data listening port*def on\_publish(client, userdata, result): *# create function for  
 # callback* print("data published to thingsboard \n")  
 pass  
  
client1 = paho.Client("iiscSmartSwitch") *# create client object*client1.on\_publish = on\_publish *# assign function to  
# callback*client1.username\_pw\_set(ACCESS\_TOKEN) *# access token from  
# thingsboard device*client1.connect(broker, port, keepalive=60) *# establish connection*while True:  
 payload = {"Humidity":60,"Temperature":25"}  
 ret = client1.publish("v1/devices/me/telemetry", payload) *# topic-  
 # v1/devices/me/telemetry* print("Please check LATEST TELEMETRY field of your device")  
 print(payload)  
 time.sleep(5)

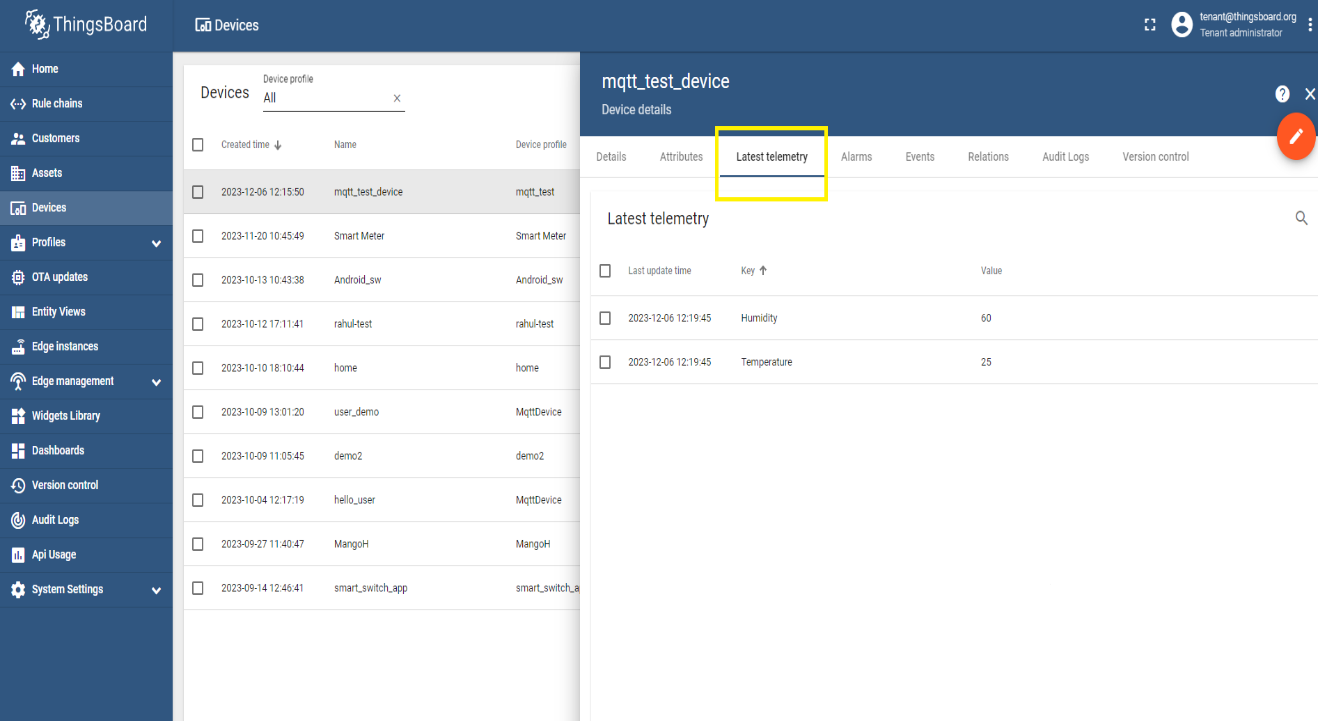
Copy access token to connect mqtt.



**1**

To check publish data you have to go into device telemetry –

\*only send data into json format only.



**1**

**2**

**3**