



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«МИРЭА – Российский технологический университет»

РТУ МИРЭА

Институт Информационных технологий

Кафедра Математического обеспечения и стандартизации информационных
технологий

Отчет по практическим работам № 9-12

по дисциплине «Технологические основы Интернета вещей»

Выполнил:

Студент группы ИНБО-08-22

Самойлов М.М.

Проверил:

Синицин И. В.

2024 г.

ОГЛАВЛЕНИЕ

ПРАКТИЧЕСКАЯ РАБОТА 9.....	3
ПРАКТИЧЕСКАЯ РАБОТА 10.....	6
ПРАКТИЧЕСКАЯ РАБОТА 11.....	10
ПРАКТИЧЕСКАЯ РАБОТА 12.....	13
ВЫВОД.....	17

ПРАКТИЧЕСКАЯ РАБОТА 9

Часть 1. Регистрация на платформе ThingsBoard

ThingsBoard имеет тестовый сервер в сборке Community Edition для проверки доступных функций платформы и тестирования своих приложений. Для регистрации на платформе необходимо перейти по данной ссылке <https://demo.thingsboard.io/signup>.

Часть 2. Создание виртуальных устройств в облаке

Согласно варианту создайте в облаке виртуальные устройства для получения данных. Каждое устройство должно иметь свой профиль, соответствующий передаваемым на устройство данным. В качестве протокола для профилей устройств используйте MQTT. Остальные параметрами оставьте незаполненными.

№ варианта	Датчики
1	1. Датчик температуры 2. Датчик движения 3. Датчик напряжения
2	1. Датчик шума 2. Датчик освещенности 3. Датчик напряжения
3	1. Датчик шума 2. Датчик качества воздуха 3. Датчик напряжения
4	1. Датчик движения 2. Датчик температуры 3. Датчик напряжения
5	1. Датчик качества воздуха 2. Датчик освещенности 3. Датчик напряжения
6	1. Датчик влажности 2. Датчик шума 3. Датчик напряжения
7	1. Датчик влажности 2. Датчик температуры 3. Датчик напряжения

Часть 3. Отправка данных в облако

Выполните передачу тестовых данных в каждое из созданных устройств. Данные должны соответствовать типу устройства, то есть, к примеру, в термометр должна поступить температура. Данные можно передавать при помощи утилиты `mosquitto_pub`. Ссылка на документацию по передаче данных на устройства по MQTT – <https://thingsboard.io/docs/reference/mqtt-api/>.

В отчете необходимо отразить созданные устройства, процесс отправки данных с облако, а также отображение этих данных на виртуальных устройствах в облачной платформе.

Рисунок 9.1 - Условие и вариант работы

Часть 1. Регистрация на платформе ThingsBoard

Данные для авторизации:

Логин: ixbo-x-21-номер_подгруппы@tb.org (ikbo-14-21-1@tb.org)

Пароль: 123123

Часть 2. Создание виртуальных устройств в облаке

Согласно выбранному варианту (Вариант 1), необходимо описать следующие исполнительные устройства и датчики:

1. Датчик температуры;
2. Датчик движения;
3. Датчик напряжения.

Для выполнения этой части практической работы нами были созданы в облаке 3 устройства (Рисунок 9.2 - 9.3):

- Датчик температуры;
- Датчик движения;
- Датчик напряжения.

Рисунок 9.2 - Создание одного девайса

<input type="checkbox"/> Врекл создан	Name	Profile type	Transport type	Description	Default	
<input type="checkbox"/>	2022-11-15 18:42:34 Датчик напряжения	Default	MQTT		<input type="checkbox"/>	
<input type="checkbox"/>	2022-11-15 18:42:44 Датчик движения	Default	MQTT		<input type="checkbox"/>	
<input type="checkbox"/>	2022-11-15 18:41:06 Датчик температуры	Default	MQTT		<input type="checkbox"/>	
<input type="checkbox"/>	2022-11-08 18:10:33 default	Default	Default	Default device profile	<input checked="" type="checkbox"/>	

Рисунок 9.3 - Созданные девайсы

Часть 3 Отправка данных в облако

Далее мы отправили данные на созданные устройства при помощи утилиты `mosquito_pub` с помощью команды (Рисунок 9.4 – 9.6).

```
user@wiredboard-AB4BCCUU:~$ mosquitto_pub -d -q 1 -h "thingsboard.mosquitto" -t "v1/devices/me/telemetry" -u "1AFpZTxaMNovtmUHL10" -m '{"temperature":19}'
Client mosqpub|10838-wiredboard sending CONNECT
Client mosqpub|10838-wiredboard received CONNACK
Client mosqpub|10838-wiredboard sending PUBLISH (d0, q1, r0, m1, 'v1/devices/me/telemetry', ... (16 bytes))
Client mosqpub|10838-wiredboard received PUBACK (Mid: 1)
Client mosqpub|10838-wiredboard sending DISCONNECT
```

Рисунок 9.4 - Отправка информации на датчик температуры

```
user@wiredboard-AB4BCCUU:~$ mosquitto_pub -d -q 1 -h "thingsboard.mosquitto" -t "v1/devices/me/telemetry" -u "QuYVRWaHyfXmewHxagFk" -m '{"move":400}'
Client mosqpub|10585-wiredboard sending CONNECT
Client mosqpub|10585-wiredboard received CONNACK
Client mosqpub|10585-wiredboard sending PUBLISH (d0, q1, r0, m1, 'v1/devices/me/telemetry', ... (10 bytes))
Client mosqpub|10585-wiredboard received PUBACK (Mid: 1)
Client mosqpub|10585-wiredboard sending DISCONNECT
```

Рисунок 9.5 - Отправка информации на датчик движения

```
user@wiredboard-AB4BCCUU:~$ mosquitto_pub -d -q 1 -h "thingsboard.mosquitto" -t "v1/devices/me/telemetry" -u "7S2Ihki7xVwViKmZUE5" -m '{"voltage":10}'
Client mosqpub|9562-wiredboard sending CONNECT
Client mosqpub|9562-wiredboard received CONNACK
Client mosqpub|9562-wiredboard sending PUBLISH (d0, q1, r0, m1, 'v1/devices/me/telemetry', ... (12 bytes))
Client mosqpub|9562-wiredboard received PUBACK (Mid: 1)
Client mosqpub|9562-wiredboard sending DISCONNECT
```

Рисунок 9.6 - Отправка информации на датчик напряжения

Устройства

Device profile

All

X

Время создания	Название	Device profile
2022-11-15 18:43:34	Датчик напряжения	Датчик напряжения
2022-11-15 18:42:44	Датчик движения	Датчик движения
2022-11-15 18:41:06	Датчик температуры	Датчик температуры

Датчик температуры

Подробности об устройстве

Details

Атрибуты

Последняя телеметрия

Оповещения

События

Отношения

Логи аудита

Version control

Последняя телеметрия

Последнее обновление	Ключ	Значение
2022-11-15 18:57:47	temperature	19

Рисунок 9.7 - Отправка значений температуры

Устройства

Device profile

All

X

Время создания	Название	Device profile
2022-11-15 18:43:34	Датчик напряжения	Датчик напряжения
2022-11-15 18:42:44	Датчик движения	Датчик движения
2022-11-15 18:41:06	Датчик температуры	Датчик температуры

Датчик движения

Подробности об устройстве

Details

Атрибуты

Последняя телеметрия

Оповещения

События

Отношения

Логи аудита

Version control

Последняя телеметрия

Последнее обновление	Ключ	Значение
2022-11-15 18:56:36	move	400

Рисунок 9.8 - Отправка значений движения

Устройства

Device profile

All

X

Время создания	Название	Device profile
2022-11-15 18:43:34	Датчик напряжения	Датчик напряжения
2022-11-15 18:42:44	Датчик движения	Датчик движения
2022-11-15 18:41:06	Датчик температуры	Датчик температуры

Датчик напряжения

Подробности об устройстве

Details

Атрибуты

Последняя телеметрия

Оповещения

События

Отношения

Логи аудита

Version control

Последняя телеметрия

Последнее обновление	Ключ	Значение
2022-11-15 18:51:26	voltage	10

Рисунок 9.9 - Отправка значений напряжения

ПРАКТИЧЕСКАЯ РАБОТА 10

Задание практической работы №10

Реализуйте скрипты из практической работы №3 при помощи цепочек правил ThingsBoard, используя приведенную в методичке инструкцию.

Примечание. При ответе на RPC запрос необходимо посылать в облако состояние физического устройства, например, состояние шарового крана, в виде JSON строки: “{“valve_state”: 1}”. Или же установившийся цвет RGB ленты: “{“rgb_color”: “255;0;0”}”.

№ варианта	Сценарии
1	1. Включение и выключение воды по датчику движения 2. Включение и выключение диодной ленты по кнопке
2	1. Включение и выключения вентилятора по концентрации CO ₂ 2. Включение, выключение и изменение звукового сигнала по кнопкам. Например, на одну кнопку значение повышается, на другую понижается, на третью происходит включение/выключение.
3	1. Изменение цвета диодной ленты по концентрации CO ₂ (зеленый цвет – концентрация в норме, красный – повышена)

	2. Включение и выключения световых индикаторов по кнопкам
4	1. Включение и выключение вентилятора по температуре 2. Открытие и закрытие шарового крана при одновременном нажатии двух кнопок
5	1. Включение и выключение вентилятора по датчику движения 2. Включение и выключения индикации зеленым и красным светом комбинированного датчика по кнопкам
6	1. Включение и выключение звукового сигнала по датчику силы тока 2. Включение и изменение цвета диодной ленты по кнопкам
7	1. Включение и выключение диодной ленты по датчику силы тока 2. Включение и выключение вентилятора при одновременном нажатии двух кнопок

В отчете необходимо отразить созданные устройства, цепочки правил, а также процесс проверки правил при помощи утилит mosquito.

Рисунок 10.1 - Условие и вариант работы

Сценарии для индивидуального варианта:

1. Включение и выключение воды по датчику движения;
2. Включение и выключения диодной ленты по кнопке.

Реализация первого сценария:

Построим цепочку (Рисунок 10.2) и начнем настраивать (Рисунок 10.3).

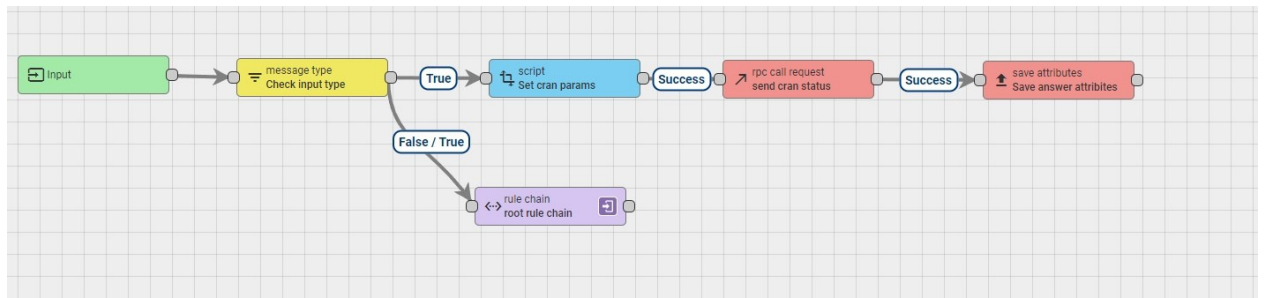


Рисунок 10.2 - Цепочка правил

Настроим профиль устройств (Рисунок 10.3):

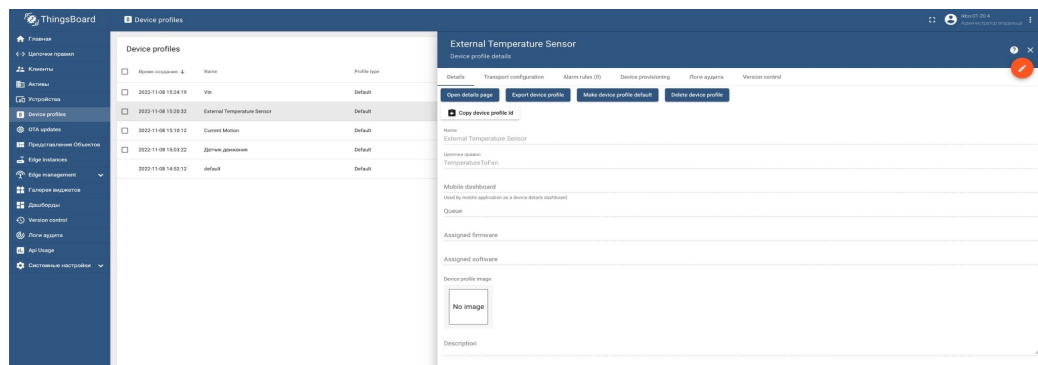


Рисунок 10.3 - Настройка профиля устройств

Напишем и протестируем скрипт (Рисунок 10.4).

```
function Transform(msg, metadata, msgType) {
```

```

1  function getNewMotionStatus(motion) {
2    return motion > 1000;
3  }
4  let newMsg = {};
5  let newMsgType = {};
6  newMsg = {
7    "method": "setValveState",
8    "params": {
9      "state": getNewMotionStatus(msg.motion)
10   }
11 };
12 newMsgType = "POST_ATTRIBUTES_REQUEST";
13 return {msg: newMsg, metadata: metadata, msgType: newMsgType};

```

Рисунок 10.4 - Скрипт

Проведем подписку на устройство и отправим данные, затем отправим ответ на запрос и проверим его в клиентских атрибутах (Рисунок 10.5-7).


```
C:\Program Files\mosquitto>mosquitto_pub -d -q 1 -h "thingsboard.mosquitto" -t "v1/devices/me/telemetry" -u "F39JivSM30Tq2cQVKNc2" -m '{"motion": 1100}'
Client null sending CONNECT
Client null received CONNACK (0)
Client null sending PUBLISH (d0, q1, r0, m1, 'v1/devices/me/telemetry', ... (14 bytes))
Client null received PUBACK (Mid: 1, RC:0)
Client null sending DISCONNECT

C:\Program Files\mosquitto>mosquitto_pub -d -q 1 -h "thingsboard.mosquitto" -t "v1/devices/me/rpc/response/13" -u "F39JivSM30Tq2cQVKNc2" -m '{"status": 1}'
Client null sending CONNECT
Client null received CONNACK (0)
Client null sending PUBLISH (d0, q1, r0, m1, 'v1/devices/me/rpc/response/13', ... (11 bytes))
Client null received PUBACK (Mid: 1, RC:0)
Client null sending DISCONNECT
```

Рисунок 10.5 – Отправленные данные

```
C:\Program Files\mosquitto>mosquitto_sub -v -h "thingsboard.mosquitto" -t "v1/devices/me/rpc/request/+" -u "F39JivSM30Tq2cQVKNc2"
v1/devices/me/rpc/request/7 {"method":"setValveState","params":{"state":true}}
v1/devices/me/rpc/request/8 {"method":"setValveState","params":{"state":true}}
v1/devices/me/rpc/request/8 {"method":"setValveState","params":{"state":true}}
v1/devices/me/rpc/request/9 {"method":"setValveState","params":{"state":false}}
v1/devices/me/rpc/request/9 {"method":"setValveState","params":{"state":false}}
v1/devices/me/rpc/request/10 {"method":"setValveState","params":{"state":true}}
v1/devices/me/rpc/request/8 {"method":"setValveState","params":{"state":true}}
v1/devices/me/rpc/request/9 {"method":"setValveState","params":{"state":false}}
v1/devices/me/rpc/request/11 {"method":"setValveState","params":{"state":true}}
v1/devices/me/rpc/request/10 {"method":"setValveState","params":{"state":true}}
v1/devices/me/rpc/request/12 {"method":"setValveState","params":{"state":true}}
v1/devices/me/rpc/request/13 {"method":"setValveState","params":{"state":true}}
```

Рисунок 10.6 – Подписка на устройство

Кран			?	×
Подробнее об устройстве				
Details	Атрибуты	Последняя телеметрия	Оповещения	События
		Отношения	Логи аудита	Version control
Клиентские атрибуты		Контекст атрибутов объекта		
		Клиентские атрибуты		
<input type="checkbox"/>	Последнее обновление	Ключ ↑	Значение	
<input type="checkbox"/>	2022-11-22 18:29:38	status	1	

Рисунок 10.7 – Клиентские атрибуты

Реализация второго сценария:

Строим цепочку (Рисунок 10.8).

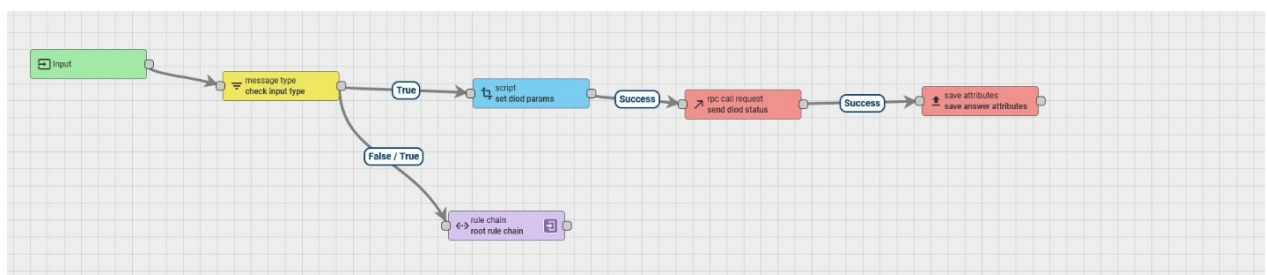


Рисунок 10.8 - Цепочка правил

Реализуем скрипт (Рисунок 10.9).


```
function Transform(msg, metadata, msgType) {
```

```
1 function getNewButtonStatus(button) {
2   return button == 1;
3 }
4 var newMsg = {};
5 var newMsgType = {};
6 newMsg = {
7   "method": "setValveState",
8   "params": {
9     "state": getNewButtonStatus(msg.button)
10  }
11 };
12 newMsgType = "POST_ATTRIBUTES_REQUEST";
13 return {msg: newMsg, metadata: metadata, msgType: newMsgType};
```

Рисунок 10.9 - Скрипт

Проводим отправку и проверим получение атрибутов (Рисунок 10.10-12).

```
C:\Program Files\mosquitto>mosquitto_pub -d -q 1 -h "thingsboard.mosquitto" -t "v1/devices/me/telemetry" -u "43PIVP8iefqEzLkTA1w3" -m '{"button": 1}'
Client null sending CONNECT
Client null received CONNACK (0)
Client null sending PUBLISH (d0, q1, r0, m1, 'v1/devices/me/telemetry', ... (11 bytes))
Client null received PUBACK (Mid: 1, RC:0)
Client null sending DISCONNECT

C:\Program Files\mosquitto>mosquitto_pub -d -q 1 -h "thingsboard.mosquitto" -t "v1/devices/me/rpc/response/0" -u "43PIVP8iefqEzLkTA1w3" -m '{"status": 1}'
Client null sending CONNECT
Client null received CONNACK (0)
Client null sending PUBLISH (d0, q1, r0, m1, 'v1/devices/me/rpc/response/0', ... (11 bytes))
Client null received PUBACK (Mid: 1, RC:0)
Client null sending DISCONNECT

C:\Program Files\mosquitto>mosquitto_pub -d -q 1 -h "thingsboard.mosquitto" -t "v1/devices/me/telemetry" -u "43PIVP8iefqEzLkTA1w3" -m '{"button": 0}'
Client null sending CONNECT
Client null received CONNACK (0)
Client null sending PUBLISH (d0, q1, r0, m1, 'v1/devices/me/telemetry', ... (11 bytes))
Client null received PUBACK (Mid: 1, RC:0)
Client null sending DISCONNECT

C:\Program Files\mosquitto>mosquitto_pub -d -q 1 -h "thingsboard.mosquitto" -t "v1/devices/me/rpc/response/0" -u "43PIVP8iefqEzLkTA1w3" -m '{"status": 0}'
Client null sending CONNECT
Client null received CONNACK (0)
Client null sending PUBLISH (d0, q1, r0, m1, 'v1/devices/me/rpc/response/0', ... (11 bytes))
Client null received PUBACK (Mid: 1, RC:0)
Client null sending DISCONNECT
```

Рисунок 10.10 – Отправка данных

```
C:\Program Files\mosquitto>mosquitto_sub -v -h "thingsboard.mosquitto" -t "v1/devices/me/rpc/request/+" -u "43PIVP8iefqEzLkTA1w3"
v1/devices/me/rpc/request/0 {"method":"setValveState","params":{"state":true}}
v1/devices/me/rpc/request/1 {"method":"setValveState","params":{"state":false}}
```

Рисунок 10.11 – Подписка на устройство

Details	Атрибуты	Последняя телеметрия	Оповещения	События	Отношения	Логи аудита	Version control
<div> <div>Клиентские атрибуты</div> <div> <div>Контекст атрибутов объекта</div> <div>Клиентские атрибуты</div> </div> </div>							
<input type="checkbox"/>	Последнее обновление	Ключ ↑	Значение				
<input type="checkbox"/>	2022-11-22 18:56:36	rgb_color	0,0,0				

Рисунок 10.12 – Клиентские атрибуты

ПРАКТИЧЕСКАЯ РАБОТА 11

Добавьте в цепочки правил из 10 практической работы формирование нескольких типов тревог:

- Для первой цепочки из варианта – тревогу при выходе приходящего параметра за допустимые границы (границы задайте самостоятельно);
- Для второй цепочки из варианта – тревогу при отсутствии ожидаемого параметра в приходящем сообщении (к примеру, в приходящем сообщении отсутствует параметр с состоянием кнопки)
- Для обеих цепочек – тревога при поступлении неверного ответа от физического устройства (это может быть неверный формат или неверное значение, отличающееся от ожидаемого).

В отчет включите обновленные цепочки правил, скрипты проверок и формирования тревог, а также результаты тестирования цепочек при помощи утилит mosquitto.

Рисунок 11.1 - Условие задания

Кран

Измененная цепочка правил (Рисунок 11.2)

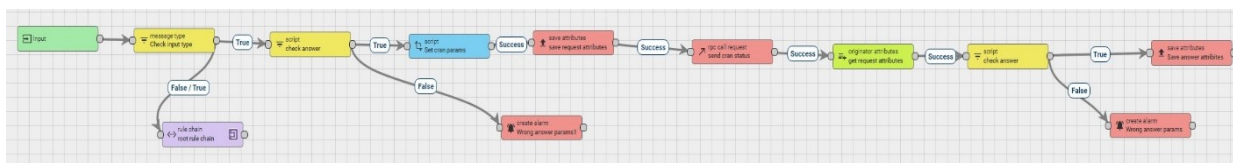


Рисунок 11.2 - Цепочка правил

Напишем скрипты с заданными условиями (Рисунок 11.3 – 11.4)

Проведем подписку и отправим значение вне рамок - получим выход за границу (Рисунок 11.5-6):

```
1 return msg.motion < 1500 && msg.motion > 500;
```

Рисунок 11.3 – Первый скрипт

```
1 var details = {};  
2 details.param_status = false;  
3 return details;
```

Рисунок 11.4 - Второй скрипт

```
C:\Program Files\mosquitto>mosquitto_pub -d -q 1 -h "thingsboard.mosit" -t "v1/devices/me/telemetry" -u "F39JIvSM3OTqZcQVKnc2" -m '{"motion": 1200}'  
Client null sending CONNECT  
Client null received CONNACK (0)  
Client null sending PUBLISH (d0, q1, r0, m1, 'v1/devices/me/telemetry', ... (14 bytes))  
Client null received PUBACK (Mid: 1, RC:0)  
Client null sending DISCONNECT  
  
C:\Program Files\mosquitto>mosquitto_pub -d -q 1 -h "thingsboard.mosit" -t "v1/devices/me/rpc/response/9" -u "F39JIvSM3OTqZcQVKnc2" -m '{"state": false}'  
Client null sending CONNECT  
Client null received CONNACK (0)  
Client null sending PUBLISH (d0, q1, r0, m1, 'v1/devices/me/rpc/response/9', ... (16 bytes))  
Client null received PUBACK (Mid: 1, RC:0)  
Client null sending DISCONNECT
```

Рисунок 11.5 - Запрос

Время создания	Индикатор	Тип	Уровень	Статус	Под...
2022-11-29 19:02:17	Кран	Wrong Answer Alarm	Критический	Активные неподтвержденные	...
2022-11-29 18:40:19	Кран	Wrong Answer Alarm	Критический	Сброшенные подтвержденные	...

Рисунок 11.6 - Оповещение о выходе за предельно допустимые границы

Далее необходимо добавить проверку на состояние и оповещение (Рисунки 11.7 – 11.10).

```
1 var request_params = JSON.parse(metadata.ss_params);
2 return msg.state === request_params.state;
```

Рисунок 11.7 – Первый скрипт

```
1 var details = {};
2 var request_params = JSON.parse(metadata.ss_params);
3 if (metadata.prevAlarmDetails) {
4   details = JSON.parse(metadata.prevAlarmDetails);
5   // Удаление поля prevAlarmDetails из метаданных
6   delete metadata.prevAlarmDetails;
7   // Теперь метаданные содержат только данные, которые были на входе
8 }
9 details.send_status = request_params.state;
10 details.answer_status = msg.state;
11 return details;
```

Рисунок 11.8 – Второй скрипт

```
C:\Program Files\mosquitto>mosquitto_pub -d -q 1 -h "thingsboard.mosit" -t "v1/devices/me/telemetry" -u "F39JIVSM30TqZcQVKnc2" -m '{"motion": 1200}'
Client null sending CONNECT
Client null received CONNACK (0)
Client null sending PUBLISH (d0, q1, r0, m1, 'v1/devices/me/telemetry', ... (14 bytes))
Client null received PUBACK (Mid: 1, RC:0)
Client null sending DISCONNECT

C:\Program Files\mosquitto>mosquitto_pub -d -q 1 -h "thingsboard.mosit" -t "v1/devices/me/rpc/response/9" -u "F39JIVSM30TqZcQVKnc2" -m '{"state": false}'
Client null sending CONNECT
Client null received CONNACK (0)
Client null sending PUBLISH (d0, q1, r0, m1, 'v1/devices/me/rpc/response/9', ... (16 bytes))
Client null received PUBACK (Mid: 1, RC:0)
Client null sending DISCONNECT
```

Рисунок 11.9 – Отправленные данные

Время создания	Индикатор	Тип	Уровень	Статус	Под...
2022-11-29 19:02:17	Кран	Wrong Answer Alarm	Критический	Активные неподтвержденные	...
2022-11-29 18:40:19	Кран	Wrong Answer Alarm	Критический	Сброшенные подтвержденные	...

Рисунок 11.10 – Полученная ошибка

Включение и выключение диодной ленты

```

graph LR
    Input[Input] --> MsgType[message type  
Check if type]
    MsgType -- True --> StripCheck[strip  
check answer]
    StripCheck -- True --> StripParam[strip  
set con param]
    StripParam -- Success --> SaveInit[save initial answer]
    SaveInit -- Success --> SetRef[set ref count  
and con status]
    SetRef -- Success --> LogInit[log initial answer  
get request param]
    LogInit -- Success --> StripCheck2[strip  
check answer]
    StripCheck2 -- True --> SendAll[send all things  
save answer param]
    StripCheck2 -- False --> CreateAlarm2[create alarm  
saving answer param]
    StripCheck -- False --> CreateAlarm1[create alarm  
saving answer param]
    MsgType -- False / True --> CreateAlarm1
  
```

Проведем подписку и отправим невалидный запрос (Рисунок 11.12-13):

```
C:\Program Files\mosquitto>mosquitto_pub -d -q 1 -h "thingsboard.mosit" -t "v1/devices/me/telemetry" -u "43P1VP8iefqEzLkTA1w3" -m "{}"
Client null sending CONNECT
Client null received CONNACK (0)
Client null sending PUBLISH (d0, q1, r0, m1, 'v1/devices/me/telemetry', ... (2 bytes))
Client null received PUBACK (Mid: 1, RC:0)
Client null sending DISCONNECT
```

Диодная лента

Подробнее об устройстве

Details

Атрибуты

Последняя телеметрия

Оповещения

События

Отношения

Логг аудита

Version control

Статус оповещения

Все

Последние 30 дней

Время создания

Индикатор

Тип

Уровень

Статус

Под...

2022-11-29 19:15:19

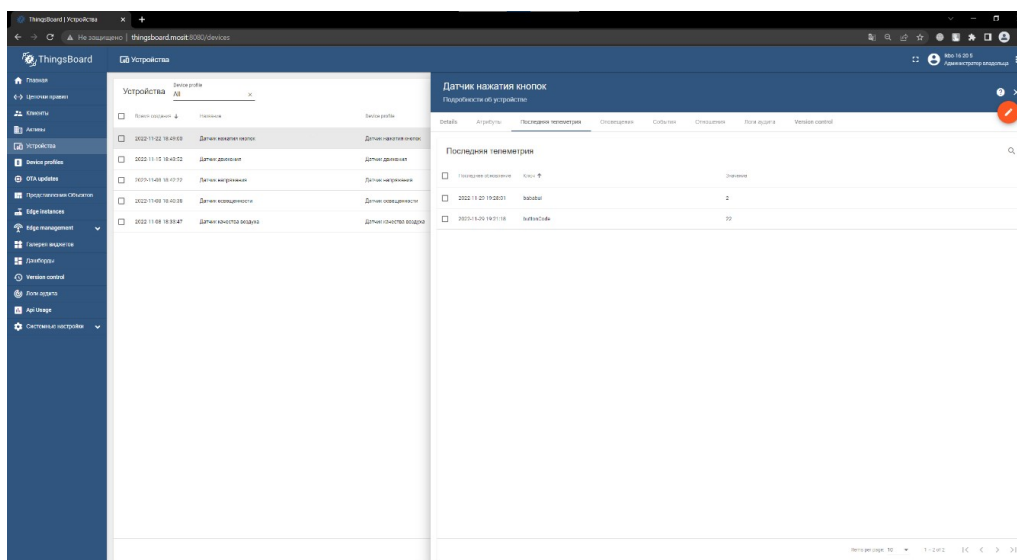
диодная лента

Wrong Answer Alarm

Критический

Активные и неподтвержденные

В случае, если все данные введены корректно, ошибку не получим (Рисунок 11.14).



12

ПРАКТИЧЕСКАЯ РАБОТА 12

Реализуйте отправку Email сообщений из облачной платформы при возникновении тревог на узлах, созданных в практической работе №11. В качестве SMTP сервера для пересылки сообщений предлагается использовать Yandex и Google, поскольку они были протестированы. Однако, можно использовать любой другой понравившийся сервис.

В отчет включите обновленные цепочки правил, параметры узлов пересылки сообщений, а также результаты тестирования цепочек при помощи утилит `mosquito` и скриншоты приходящих электронных писем.

Рисунок 12.1 - Условие и вариант работы

Для почтового сервиса выбрана Yandex почта

Изменим цепочку правил для диодной ленты (Рисунок 12.2)

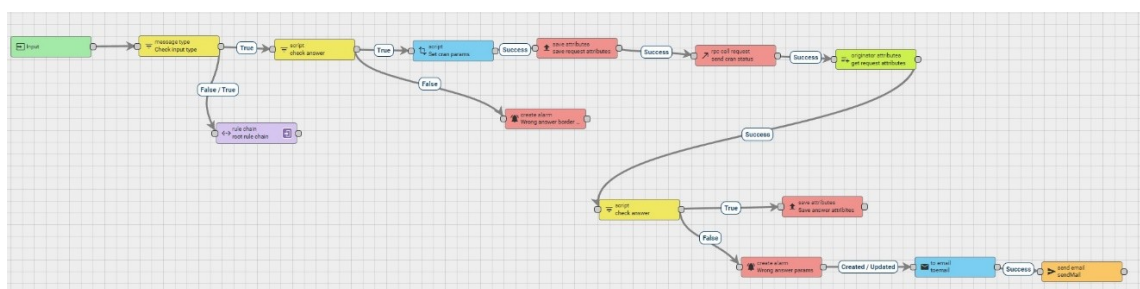


Рисунок 12.2 - Цепочка правил

Произведем настройки блока to email (Рисунок 12.3)

toemail

Преобразование - to email

Подробности

События

Помощь

Название *

toemail

From Template *

kosgor2001@yandex.ru

Hint: use \${metadatakey} for value from metadata, \${messagekey} for value from message body

To Template *

kosgor2001@yandex.ru

Comma separated address list, use \${metadatakey} for value from metadata, \${messagekey} for value from message body

Cc Template

Comma separated address list, use \${metadatakey} for value from metadata, \${messagekey} for value from message body

Bcc Template

Comma separated address list, use \${metadatakey} for value from metadata, \${messagekey} for value from message body

Subject Template *

Device alarm

Hint: use \${metadatakey} for value from metadata, \${messagekey} for value from message body

Mail body type

Plain Text

Body Template *

Device alarm

Hint: use \${metadatakey} for value from metadata, \${messagekey} for value from message body

Описание

Рисунок 12.3 – Данные почты

Отправим данные и получим письмо на указанную в цепочке правил

почту (Рисунок 12.4-6).

```
C:\Program Files\mosquitto>mosquitto_pub -d -q 1 -h "thingsboard.mosquitto" -t "v1/devices/me/telemetry" -u "43PIVP8iefqEzLkTA1w3" -m "{\"button\": 0}"
Client null sending CONNECT
Client null received CONNACK (0)
Client null sending PUBLISH (d0, q1, r0, m1, 'v1/devices/me/telemetry', ... (11 bytes))
Client null received PUBACK (Mid: 1, RC:0)
Client null sending DISCONNECT

C:\Program Files\mosquitto>mosquitto_pub -d -q 1 -h "thingsboard.mosquitto" -t "v1/devices/me/rpc/response/7" -u "43PIVP8iefqEzLkTA1w3" -m "{\"state\": true}"
Client null sending CONNECT
Client null received CONNACK (0)
Client null sending PUBLISH (d0, q1, r0, m1, 'v1/devices/me/rpc/response/7', ... (15 bytes))
Client null received PUBACK (Mid: 1, RC:0)
Client null sending DISCONNECT
```

Рисунок 12.4 - Отправка запроса

Подобности События Помощь										
Тип события		Отладка		Последние 1 день						
Время возникновения события	Сервер	Тип	Тип объекта	Entity Id	Ид сообщения	Тип сообщения	Тип отношения	Данные	Метаданные	Ошибки
2022-12-06 18:39:35	depo-thingsboard	OUT	DEVICE	10419a...	5e049a...	SEND_EMAIL	Success	***	***	
2022-12-06 18:39:35	depo-thingsboard	IN	DEVICE	10419a...	5e049a...	SEND_EMAIL	Success	***	***	
2022-12-06 18:32:15	depo-thingsboard	OUT	DEVICE	10419a...	6b329a...	SEND_EMAIL	Failure	***	***	***
2022-12-06 18:32:05	depo-thingsboard	IN	DEVICE	10419a...	6b329a...	SEND_EMAIL	Success	***	***	

Рисунок 12.5 - Отправка письма

Изменим цепочку правил для крана (Рисунок 12.6)

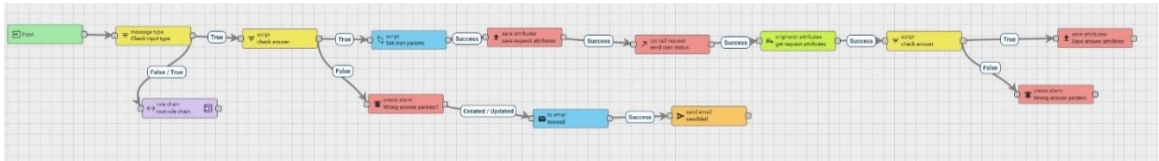


Рисунок 12.6 - Цепочка правил

Произведем настройки блока to email (Рисунок 12.7)

toemail
Преобразование - to email

Подобности
События
Помощь

Название *
toemail

From Template *
kosgor2001@yandex.ru

Hint: use \${netdatakey} for value from metadata, \${messagekey} for value from message body

To Template *
kosgor2001@yandex.ru

Comma separated address list, use \${netdatakey} for value from metadata, \${messagekey} for value from message body

Cc Template

Comma separated address list, use \${netdatakey} for value from metadata, \${messagekey} for value from message body

Bcc Template

Comma separated address list, use \${netdatakey} for value from metadata, \${messagekey} for value from message body

Subject Template *
Ограничение прикинено или превышено

Hint: use \${netdatakey} for value from metadata, \${messagekey} for value from message body

Mail body type
Plain Text

Body Template *
Ограничение прикинено или превышено

Hint: use \${netdatakey} for value from metadata, \${messagekey} for value from message body

Описание

Рисунок 12.7 – Данные почты

Отправим данные и получим письмо на указанную в цепочке правил почту (Рисунок 12.8-10).

```
C:\Program Files\mosquitto>mosquitto_pub -d -q 1 -h "thingsboard.mosit" -t "v1/devices/me/telemetry" -u "F39JivSM30TqZcQVKNc2" -m "{\"motion\": 400}"
Client null sending CONNECT
Client null received CONNACK (0)
Client null sending PUBLISH (d0, q1, r0, m1, 'v1/devices/me/telemetry', ... (15 bytes))
Client null received PUBACK (Mid: 1, RC:0)
Client null sending DISCONNECT
```

Рисунок 12.8 - Отправка запроса

sendMail
Сторонние - send email

Подобности
События
Помощь

Название *
sendMail

☐ Use system SMTP settings

Protocol
SMTPS

SMTP host *
smtp.yandex.ru
SMTP port *
465

Timeout ms *
10000

☐ Enable TLS

☐ Enable proxy

Username
kosgor2001@yandex.ru

Password

Описание
да

Рисунок 12.9 - Отправка письма

ВЫВОД

В ходе выполнения практических работ 9 - 12 была изучена работа с облачными платформами IoT, проведено управление устройствами с помощью платформ, налажено уведомление об изменении состоянии системы. Так же написаны цепочки правил с добавлением узлом разных типов. Углублены навыки работы со стендом Wirenboard.