

- Распределенные системы и сети
 - Урок 4. Распределённые системы
 - Домашнее задание
- Решение
 - Взаимодействие через браузер
 - ChirpStack
 - Создание API-token
 - Applications
 - ChirpStack API
 - Bearer token
 - GET
 - Responses
 - POST
 - Application id
 - Device profile id
 - DevEUI
 - Responses
 - Result
 - PUT
 - Responses
 - Result
 - DELETE
 - Responses
 - Result
 - POST Queue to real device
 - Data to Base64
 - Responses
 - Result
 - Взаимодействие с помощью Python
 - Создание 10 устройств
 - Запрос информации о 10 устройствах
 - Обновить имена 10 устройств
 - Удаление 10 устройств

Распределенные системы и сети

Урок 4. Распределённые системы

Домашнее задание

Прислать результаты выполненных запросов через Swagger UI и Postman.

Прислать скрипт на любом удобном языке программирования, где через API будут созданы 10 устройств (все переменные должны задаваться в коде, списком или массивом, но одной строкой), будет запрошена информация о 10 устройствах, будут обновлены имена 10 устройств, будут удалены 10 устройств.

Желательно использовать Python через Google Collab. Для сдачи ДЗ студент должен предоставить преподавателю ссылку с доступом к блокноту с кодом (перед отправкой ДЗ проверьте, что ссылка открывается в приватном режиме браузера).

Решение

Взаимодействие через браузер

ChirpStack

Создание API-token

ChirpStack

Search...

gb@iot.ru

GB_IoT_Tenant

Network Server

Regions

Tenant

Dashboard

Users

API Keys

Device Profiles

Gateways

Applications

Tenants / GB_IoT_Tenant / API Keys

API keys

Add API key

ID	Name	Action
c6c8ccd4-1caa-469a-a3c2-390f2527abf9	abcser	
862a33b6-becd-4b52-9a65-6d3e9f786ae3	ega-5471	
49644b19-91a7-43d3-9fe2-93662486ef6d	GVI	
251bd910-2143-408b-92ba-eb912685ac0a	GVI-1234	
00a3c805-74a6-4132-9b5d-60233b15f29a	GVI-5471	
c13990dd-327b-4fcc-be57-d35efe653251	RRG-5471	

Applications

Tenants / GB_IoT_Tenant / Applications / GB_IoT_Application

GB_IoT_Application application id: 6bb2cce7-abe9-4e98-980e-f5ada088d177

Delete application

Devices

Multicast groups

Relays

Application configuration

Integrations

Add device

Selected devices

<input type="checkbox"/>	Last seen	DevEUI	Name	Device profile	Battery
<input type="checkbox"/>	2024-02-14 08:57:34	3441a8ffe00086d	DevKit	OTAA_LoRaWAN-1.0.4_Class-C_RU864	


<

1

>

10 / page

ChirpStack API

 Swagger
Supported by SMARTBEAR

api.json

Explore

ChirpStack REST API v4.6.0

api.json

For documentation, please refer to www.chirpstack.io.


Authorize 

ApplicationService ^

GET

/api/applications


Get the list of applications.

⌵ 

POST

/api/applications


Create creates the given application.

⌵ 

PUT

/api/applications/{application.id}

Update updates the given application.

⌵ 

Bearer token

Available authorizations

✕

api_key (apiKey)

Authorized

Name: Grpc-Metadata-Authorization

In: header

Value: *****

Logout

Close

GET

DeviceService

GET

/api/devices

Get the list of devices.

Parameters

Name	Description
<div>limit</div> <div>integer(\$int64)</div> <div>(query)</div>	<div>Max number of devices to return in the result-set.</div> <div>5</div>
<div>offset</div> <div>integer(\$int64)</div> <div>(query)</div>	<div>Offset in the result-set (for pagination).</div> <div>offset</div>
<div>search</div> <div>string</div> <div>(query)</div>	<div>If set, the given string will be used to search on name (optional).</div> <div>search</div>
<div>applicationId</div> <div>string</div> <div>(query)</div>	<div>Application ID (UUID) to filter devices on.</div> <div>6bb2cce7-abe9-4e98-980e-f5ada088d177</div>

Responses

Responses

Curl

```
curl -X 'GET' \
  'https://chirpstack-api.iotserv.ru/api/devices?limit=5&applicationId=6bb2cce7-abe9-4e98-980e-f5ada088d177' \
  -H 'accept: application/json' \
  -H 'Grpc-Metadata-Authorization: Bearer eyJ0eXAI0iJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJhdWQiOiJjaGlycHN0YWNRiIiwiaXNzIj
```

Request URL

https://chirpstack-api.iotserv.ru/api/devices?limit=5&applicationId=6bb2cce7-abe9-4e98-980e-f5ada088d177

Server response

Code	Details
------	---------

200	
-----	--

Response body

```
{
  "totalCount": 1,
  "result": [
    {
      "devEui": "3441a8fffe00086d",
      "createdAt": "2024-01-17T10:50:58.793954Z",
      "updatedAt": "2024-01-17T10:50:58.793954Z",
      "lastSeenAt": "2024-02-14T08:57:34.178648Z",
      "name": "DevKit",
      "description": "LoRaWAN DevKit",
      "deviceProfileId": "29d26ba4-2bf0-452e-9341-2671f442c7da",
      "deviceProfileName": "OTAA_LoRaWAN-1.0.4_Class-C_RU864",
      "deviceStatus": {
        "margin": 6,
        "externalPowerSource": false,
        "batteryLevel": 100
      }
    }
  ]
}
```

POST

POST

/api/devices

Create the given device.

Parameters

Name	Description
<div>body ★ required</div> <div>object</div> <div>(body)</div>	<div>Edit Value Model</div> <div><pre>{ "device": { "applicationId": "6bb2cce7-abe9-4e98-980e-f5ada088d177", "description": "Device from RRG", "devEui": "0123456789ABCDEF", "deviceProfileId": "29d26ba4-2bf0-452e-9341-2671f442c7da", "isDisabled": false, "joinEui": "0000000000000000", "name": "Device from RRG", "skipFcmtCheck": false }}</pre></div> <div>Cancel</div> <div>Parameter content type</div> <div>application/json</div>

Execute

Application id

Tenants / GB_IoT_Tenant / Applications / GB_IoT_Application

GB_IoT_Application

application id: 6bb2cce7-abe9-4e98-980e-f5ada088d177

Device profile id

Tenants / GB_IoT_Tenant / Device profiles / OTAA_LoRaWAN-1.0.4_Class-C_RU864

OTAA_LoRaWAN-1.0.4_Class-C_RU864

device profile id: 29d26ba4-2bf0-452e-9341-2671f442c7da

DevEUI

Любое 64-битное значение в HEX: 0123456789abcdef

Responses

Curl

```
curl -X 'POST' \
  'https://chirpstack-api.iotserv.ru/api/devices' \
  -H 'accept: application/json' \
  -H 'Grpc-Metadata-Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ1b290eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ1b290eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.' \
  -H 'Content-Type: application/json' \
  -d '{
    "device": {
      "applicationId": "6bb2cce7-abe9-4e98-980e-f5ada088d177",
      "description": "Device from RRG",
      "devEui": "0123456789ABCDEF",
      "deviceProfileId": "29d26ba4-2bf0-452e-9341-2671f442c7da",
      "isDisabled": false,
      "joinEui": "0000000000000000",
      "name": "Device from RRG",
      "skipFcmtCheck": false
    }
  }'
```

Request URL

https://chirpstack-api.iotserv.ru/api/devices

Server response

Code	Details
200	<div><p>Response body</p><pre>{}</pre><p>Response headers</p><pre>content-length: 2 content-type: application/json date: Sat, 17 Feb 2024 10:48:29 GMT grpc-metadata-content-type: application/grpc grpc-metadata-date: Sat, 17 Feb 2024 10:48:29 GMT grpc-metadata-x-log-dev_eui: 0123456789ABCDEF grpc-metadata-x-log-is_disabled: false server: nginx x-http-version: HTTP/2.0</pre></div>

Result

Tenants / GB IoT Tenant / Applications / GB IoT Application

GB_IoT_Application application id: 6bb2cce7-abe9-4e98-980e-f5ada088d177

Devices

Multicast groups

Relays

Application configuration

Integrations

<input type="checkbox"/>	Last seen	DevEUI	Name	Device profile
<input type="checkbox"/>	Never	0123456789abcdef	Device from RRG	OTAA_LoRaWAN

PUT

Responses

Curl

```
curl -X 'PUT' \
'https://chirpstack-api.iotserv.ru/api/devices/0123456789abcdef' \
-H 'accept: application/json' \
-H 'Grpc-Metadata-Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz'
-H 'Content-Type: application/json' \
-d '{
  "device": {
    "applicationId": "6bb2cce7-abe9-4e98-980e-f5ada088d177",
    "deviceProfileId": "29d26ba4-2bf0-452e-9341-2671f442c7da",
    "name": "Device from RRG UPDATED"
  }
}'
```

Request URL

`https://chirpstack-api.iotserv.ru/api/devices/0123456789abcdef`

Server response

Code	Details
200	<div><p>Response body</p><pre>{}</pre><p>Response headers</p><pre>content-length: 2 content-type: application/json date: Sat, 17 Feb 2024 11:07:14 GMT grpc-metadata-content-type: application/grpc grpc-metadata-date: Sat, 17 Feb 2024 11:07:14 GMT grpc-metadata-x-log-dev_eui: 0123456789abcdef grpc-metadata-x-log-is_disabled: false server: nginx x-http-version: HTTP/2.0</pre></div>

Responses

Responses

Curl

```
curl -X 'PUT' \
'https://chirpstack-api.iotserver.ru/api/devices/0123456789abcdef' \
-H 'accept: application/json' \
-H 'Grpc-Metadata-Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiIsInR5cGE6YWV0IjoibnVudC5kaWQyLmFkbGkiOjEwMDEyOTkxMTYyMDAwfQ%3D' \
-H 'Content-Type: application/json' \
-d '{
  "device": {
    "applicationId": "6bb2cce7-abe9-4e98-980e-f5ada088d177",
    "deviceProfileId": "29d26ba4-2bf0-452e-9341-2671f442c7da",
    "name": "Device from RRG UPDATED"
  }
}'
```

Request URL

`https://chirpstack-api.iotserv.ru/api/devices/0123456789abcdef`

Server response

Code	Details
200	<div><p>Response body</p><pre>{}</pre><p>Response headers</p><pre>content-length: 2 content-type: application/json date: Sat, 17 Feb 2024 11:07:14 GMT grpc-metadata-content-type: application/grpc grpc-metadata-date: Sat, 17 Feb 2024 11:07:14 GMT grpc-metadata-x-log-dev_eui: 0123456789abcdef grpc-metadata-x-log-is_disabled: false server: nginx x-http-version: HTTP/2.0</pre></div>

Result

Tenants	GB	IoT Tenant	Applications	GB	IoT Application
1	10	10	10	10	10
2	10	10	10	10	10
3	10	10	10	10	10
4	10	10	10	10	10
5	10	10	10	10	10
6	10	10	10	10	10
7	10	10	10	10	10
8	10	10	10	10	10
9	10	10	10	10	10
10	10	10	10	10	10
11	10	10	10	10	10
12	10	10	10	10	10
13	10	10	10	10	10
14	10	10	10	10	10
15	10	10	10	10	10
16	10	10	10	10	10
17	10	10	10	10	10
18	10	10	10	10	10
19	10	10	10	10	10
20	10	10	10	10	10
21	10	10	10	10	10
22	10	10	10	10	10
23	10	10	10	10	10
24	10	10	10	10	10
25	10	10	10	10	10
26	10	10	10	10	10
27	10	10	10	10	10
28	10	10	10	10	10
29	10	10	10	10	10
30	10	10	10	10	10
31	10	10	10	10	10
32	10	10	10	10	10
33	10	10	10	10	10
34	10	10	10	10	10
35	10	10	10	10	10
36	10	10	10	10	10
37	10	10	10	10	10
38	10	10	10	10	10
39	10	10	10	10	10
40	10	10	10	10	10
41	10	10	10	10	10
42	10	10	10	10	10
43	10	10	10	10	10
44	10	10	10	10	10
45	10	10	10	10	10
46	10	10	10	10	10
47	10	10	10	10	10
48	10	10	10	10	10
49	10	10	10	10	10
50	10	10	10	10	10
51	10	10	10	10	10
52	10	10	10	10	10
53	10	10	10	10	10
54	10	10	10	10	10
55	10	10	10	10	10
56	10	10	10	10	10
57	10	10	10	10	10
58	10	10	10	10	10
59	10	10	10	10	10
60	10	10	10	10	10
61	10	10	10	10	10
62	10	10	10	10	10
63	10	10	10	10	10
64	10	10	10	10	10
65	10	10	10	10	10
66	10	10	10	10	10
67	10	10	10	10	10
68	10	10	10	10	10
69	10	10	10	10	10
70	10	10	10	10	10
71	10	10	10	10	10
72	10	10	10	10	10
73	10	10	10	10	10
74	10	1			

GB_IoT_Application application id: 6bb2cce7-abe9-4e98-980e-f5ada088d177

Devices

Multicast groups

Relays

Application configuration

Integrations

Ad

<input type="checkbox"/>	Last seen	DevEUI	Name	Device profile
<input type="checkbox"/>	Never	0123456789abcdef	Device from RRG UPDATED	OTAA_LoRaWAN C_RU864

DELETE

DELETE

/api/devices/{devEui}

Delete the device with the given DevEUI.

Parameters

Name	Description
devEui ★ required string (path)	DevEUI (EUI64). <div>0123456789ABCDEF</div>

Execute

Responses

Responses	
Curl	<pre>curl -X 'DELETE' \ 'https://chirpstack-api.iotserv.ru/api/devices/0123456789ABCDEF' \ -H 'accept: application/json' \ -H 'Grpc-Metadata-Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiIsInR5cGEiOiJkaWYxIiwiaWF0IjoiMTkxMTEyOTQwLmF1dG8ifQ=='</pre>
Request URL	<pre>https://chirpstack-api.iotserv.ru/api/devices/0123456789ABCDEF</pre>
Server response	
Code	Details
200	<div><p>Response body</p><pre>{}</pre></div> <div><p>Response headers</p><pre>content-length: 2 content-type: application/json date: Sat, 17 Feb 2024 11:15:32 GMT grpc-metadata-content-type: application/grpc grpc-metadata-date: Sat, 17 Feb 2024 11:15:32 GMT grpc-metadata-x-log-dev_eui: 0123456789ABCDEF server: nginx x-http-version: HTTP/2.0</pre></div>

Result

Tenants / GB_IoT_Tenant / Applications / GB_IoT_Application

GB_IoT_Application application id: 6bb2cce7-abe9-4e98-980e-f5ada088d177

Devices

Multicast groups

Relays

Application configuration

Integrations

Add device

<input type="checkbox"/>	Last seen	DevEUI	Name	Device profile
<input type="checkbox"/>	2024-02-14 08:57:34	3441a8fffe00086d	DevKit	OTAA_LoRaWAN-1.0.4_Class-C_RU864

< 1


POST Queue to real device

POST /api/devices/{queueItem.devEui}/queue Enqueue adds the given item to the downlink queue.

Parameters

Name	Description
queueItem.devEui ★ required string (path)	Device EUI (EUI64). <div>3441a8fffe00086d</div>
body ★ required object (body)	<div>Edit Value Model</div> <div><pre>{ "queueItem": { "data": "VGVzdCBmcm9tIFJSRw==", "fPort": 123 }}</pre></div>

Data to Base64



The screenshot shows the cryptii.com website. The URL bar displays `https://cryptii.com/pipes/hex-to-base64`. The navigation menu includes links for **Home**, **API**, **Privacy**, **FAQ**, **Feedback**, and **Sign in**. A sidebar on the left contains a search bar and a list of categories: **Text**, **Image**, **File**, **Audio**, **Video**, **Code**, **Network**, **Security**, **Hashing**, **Encryption**, **Decryption**, **Compression**, **Decompression**, **Encoding**, **Decoding**, **Conversion**, **Validation**, **Comparison**, **Analysis**, **Tools**, and **Utilities**. The main content area features a table with four columns, each representing a different data format or operation. The first column is labeled **Text** and contains the input `Test from RRG`. The second column is labeled **Bytes** and shows the input in hexadecimal format: `54 65 63 74 20 66 72 6f 64 20 52 52 47`. The third column is labeled **Base64** and shows the output in Base64 format: `VGZzdCBmcm9tIFJJSF`. The fourth column is labeled **Text** and shows the output in plain text: `Test from RRG`. The table also includes a **VIEW** button for each column. A sidebar on the right contains a search bar and a list of categories: **Text**, **Image**, **File**, **Audio**, **Video**, **Code**, **Network**, **Security**, **Hashing**, **Encryption**, **Decryption**, **Compression**, **Decompression**, **Encoding**, **Decoding**, **Conversion**, **Validation**, **Comparison**, **Analysis**, **Tools**, and **Utilities**. A footer at the bottom of the page contains a copyright notice: **© 2020 cryptii.com**.

VIEW	VIEW	ENCODE DECODE	VIEW
Text	Bytes	Base64	Text
Test from RRG	FORMAT: Hexadecimal GROUP BY: Byte 54 65 63 74 20 66 72 6f 64 20 52 52 47	VARIANT: Base64 (RFC 3548, RFC 4648) → Encoded 20 chars	VGZzdCBmcm9tIFJJSF

Responses

Responses

Curl

```
curl -X 'POST' \
'https://chirpstack-api.iotserver.ru/api/devices/3441a8fffe00086d/queue' \
-H 'accept: application/json' \
-H 'Grpc-Metadata-Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.' \
-H 'Content-Type: application/json' \
-d '{
    "queueItem": {
        "data": "VGZzdCBmcm9tIFJSRw==",
        "fPort": 123
    }
}'
```

Request URL

<https://chirpstack-api.iotserv.ru/api/devices/3441a8fffe00086d/queue>

Server response

Code	Details
200	<div><p>Response body</p><pre>{ "id": "5a67671b-02df-4cc7-b8bd-c07aa37df87a" }</pre></div> <div><p>Response headers</p><pre>content-length: 45 content-type: application/json date: Sat,17 Feb 2024 11:36:00 GMT grpc-metadata-content-type: application/grpc grpc-metadata-date: Sat,17 Feb 2024 11:36:00 GMT grpc-metadata-x-log-dev_eui: 3441a8fffe00086d server: nginx x-http-version: HTTP/2.0</pre></div>

Result

Tenants / GB_IoT_Tenant / Applications / GB_IoT_Application / Dev

DevKit device eui: 3441a8fffe00086d

Dashboard

Configuration

OTAA keys

Activation

Qu

2024-02-17 11:36:01

UnconfirmedDataDown

▼ phy_payload: {} 3 keys

▼ mhdr: {} 2 keys

m_type: "UnconfirmedDataDown"

major: "LoRaWANR1"

▼ mic: [] 4 items

0: 244

1: 147

2: 176

3: 76

▼ payload: {} 3 keys

f_port: 123

▼ fhdr: {} 4 keys

devaddr: "02d92827"

f_cnt: 18

▼ f_ctrl: {} 6 keys

ack: false

UART Output 2

▼ UART Output 2

Time	RSSI	SNR	COUNTER	PORT	Payload	Decode
2024-02-17 11:36:05	-45	6	18	123	546573742066726F6D20525247	Test from RRG

Взаимодействие с помощью Python

Создание 10 устройств

```
import requests
import json

token="eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJhdWQiOiJjaGlycHN0YWNRiIiwiaXNzIjoiy2hpcnBzdGFjayIsInN1YiI6ImMxMzk5MGRkLTMyN2ItNGZjYy1iZTUzLWQzNWVmZTY1MzI1MSIsInR5cCI6ImtleSJ9.irLcZ138B8-UTrYhWZG-gFXwvsDMheqz8uioza0yGzs"
url = 'https://chirpstack-api.iotserver.ru/api/devices'
applicationId = "6bb2cce7-abe9-4e98-980e-f5ada088d177"
deviceProfileId = "29d26ba4-2bf0-452e-9341-2671f442c7da"
devQuantity = 10
devicesList = [{"i:02d}23456789ABCDEF" for i in range(devQuantity)]
```

```
def create(url, token, applicationId, deviceProfileId, name, devEui):
    headers = {
        'accept': 'application/json',
        'Grpc-Metadata-Authorization': f'Bearer {token}',
        'Content-Type': 'application/json',
    }
```

```

data = {
    "device": {
        "applicationId": applicationId,
        "description": name,
        "devEui": devEui,
        "deviceProfileId": deviceProfileId,
        "isDisabled": False,
        "joinEui": "0000000000000000",
        "name": name,
        "skipFcmtCheck": False
    }
}
return requests.post(url, json=data, headers=headers)

for i in range(len(devicesList)):
    print(create(url, token, applicationId, deviceProfileId, f"RRG Device
{i:02d}", devicesList[i]))

```

Запрос информации о 10 устройствах

```

def info(url, token, applicationId, limits):
    headers = {
        'accept': 'application/json',
        'Grpc-Metadata-Authorization': f'Bearer {token}',
    }
    params = {
        'limit': limits,
        'applicationId': applicationId
    }
    response = requests.get(url, headers=headers, params=params)
    return response.json()

print(json.dumps(info(url, token, applicationId, devQuantity+1), indent=4))

```

Обновить имена 10 устройств

```

def update(url, token, applicationId, deviceProfileId, newName, devEui):
    headers = {
        'accept': 'application/json',
        'Grpc-Metadata-Authorization': f'Bearer {token}',
        'Content-Type': 'application/json',
    }
    data = {
        "device": {
            "applicationId": applicationId,
            "deviceProfileId": deviceProfileId,

```

```

        "name": newName,
        "description": newName
    }
}
put_url = f"{url}/{devEui}"
return requests.put(put_url, json=data, headers=headers)

for i in range(len(devicesList)):
    print(update(url, token, applicationId, deviceProfileId, f"RRG Device
{i:02d} UPDATED", devicesList[i]))

print(json.dumps(info(url, token, applicationId, devQuantity+1), indent=4))

```

Удаление 10 устройств

```

def delete(url, token, devEui):
    headers = {
        'accept': 'application/json',
        'Grpc-Metadata-Authorization': f'Bearer {token}',
    }
    return requests.delete(f"{url}/{devEui}", headers=headers)

for i in range(len(devicesList)):
    print(delete(url, token, devicesList[i]))

```