

Senseway Mission Connect

REST API 仕様書

2019/11/15 Ver. 0.94

履歴

日時	バージョン	内容
2019/06/19	Ver. 0.9	初版
2019/07/03	Ver. 0.91	最新データ取得 API 追加、データ送信 API 追加
2019/07/10	Ver. 0.92	クエリパラメータ from の説明追加
2019/10/24	Ver. 0.93	データ取得 API 仕様変更
2019/11/15	Ver. 0.94	データ取得用設定画面追加

目次

1 はじめに.....	1
1.1 データ取得.....	1
2 共通.....	2
2.1 使い方概要.....	2
2.2 ステータスコード.....	2
2.3 JSON フォーマット.....	2
2.4 成功時レスポンス.....	3
2.5 失敗時レスポンス.....	3
2.6 実行例.....	4
3 全 API 一覧.....	5
3.1 接続先.....	5
3.2 全 API 一覧.....	5
4 API リファレンス.....	6
4.1 ログイン.....	6
4.2 ユーザ情報取得.....	9
4.3 デバイス一覧.....	12
4.4 デバイス情報取得.....	16
4.5 GW デバイス一覧.....	20
4.6 GW デバイス情報取得.....	24
4.7 データデバイス一覧.....	27
4.8 データ取得.....	29
4.9 最新データ取得.....	36
4.10 データ送信.....	40

1 はじめに

本 API は、Senseway Mission Connect における LoRaWAN デバイスの登録情報や送信データ内容を外部からプログラム等で取得したり、デバイスに対してデータを送信するための HTTP REST API である。

1.1 データ取得

データ取得 API でデバイスから送信されたデータを取得するためには、データを送信する前に Senseway Mission Connect の管理画面にログインして、[デバイス管理]から該当デバイスの[編集]で、「REST API」を「使用する」に設定しておく必要がある。(デフォルト設定は「使用しない」)

The screenshot shows the 'SenseWay Mission Connect' management interface. The 'デバイス管理' (Device Management) tab is active. The '編集' (Edit) page for a device is displayed. The 'REST API' setting is highlighted with a red box, showing '使用する' (Use) selected. The '更新' (Update) button at the bottom left is also highlighted with a red box.

項目	設定値
DevEUI	00000b278ad9f712
AppKey	f87fe03d3e91f0487a0576b0ab17583d
AppEUI	abcdef0123456789
タイプ	Kiwitech ADB9225
ステータス	使用中
REST API	<input type="radio"/> 使用しない <input checked="" type="radio"/> 使用する
名前	Meeting Room 1 Door Sensor
備考	会議室1 入りロドア開閉センサー
備考2	
外部連携設定	なし
外部連携最終結果	

2 共通

以下に、全APIで共通の内容を記載する。

2.1 使い方概要

ユーザ名、パスワードを渡してログインAPIを呼び出すことにより、Mission Connect APIにログインしてAPIトークンを取得することができる。以降は、取得したAPIトークンをAuthorizationヘッダにBearerとして指定することにより、他のAPIが使用できる。

APIトークンには有効期間があり、期限切れになると以降のAPI呼び出しで401エラーが返る。その場合、再ログインしてAPIトークンを再取得する必要がある。

(同一アカウントでこのAPIを利用する別のプロセスが動作している場合、別のプロセスが期限切れを判定して再ログインを行うと、そのアカウントのAPIトークンが更新されるため、期限切れではなく認証エラーとなる(ステータスコードは同じ401)。この場合でも同じように再ログイン処理を行えばよい。)

2.2 ステータスコード

Mission Connect APIは、以下のステータスコードでレスポンスを返す可能性がある。

API 毎には返らないものも含まれるが、プログラム全体で処理する場合は以下を考慮する必要がある。

コード	意味
200	成功
400	不正リクエスト (リソース, パラメータエラー等)
401	認証失敗 (未認証, APIトークン期限切れ等)
402	支払い要求 (ユーザ強制停止中)
404	対象が存在しない
405	無効メソッド
500	サーバエラー
501	未実装

2.3 JSON フォーマット

リクエストボディとレスポンスはJSONフォーマットで、本仕様書での表記は人間が読みやすいように整形しているが、実際は以下のようなインデント/改行なしフォーマットである。

```
{"result": true, "status": 200, "devices": [ "0000624266e31024", "00008032b2a22000"]}
```

2.4 成功時レスポンス

API 呼び出しに成功した場合は、以下のレスポンスを返す。

{各 API ごとの内容}の部分は、各 API ごとに異なる内容が入る。

```
{
  "result": true,
  "status": 200,
  {各 API ごとの内容}
  .
}
```

キー名	型	概要
result	boolean	true (成功)
status	number	ステータスコード (200 = 成功)
以降は、各 API ごとの内容が含まれる。		

2.5 失敗時レスポンス

API 呼び出しに失敗した場合は、以下のレスポンスを返す。

詳細な理由がある場合は、エラーメッセージだけでなく、{詳細エラーメッセージ}が追加される。これは、複数の理由が含まれる場合があるので配列となっている。

```
{
  "result": false,
  "status": {ステータスコード},
  "message": {エラーメッセージ},
  "details": [
    {詳細エラーメッセージ},
    .
    (繰り返し)
  ]
}
```

キー名	型	概要
result	boolean	false (失敗)
status	number	ステータスコード
message	string	エラーメッセージ
details (ある場合)	string (配列)	詳細エラーメッセージ 詳細な理由がある場合は、エラーメッセージだけでなく、詳細エラーメッセージ(複数の可能性あり)が追加される。

2.6 実行例

実行例では、Linux 上の shell にて、API 発行に curl コマンド、JSON 整形用に jq コマンドを使用している。(いくつかの不要な表示は省略している。)

```
#
# パスワードが異なりログイン失敗
#

$ curl -k -X POST -H 'Content-Type: application/json' -H 'Accept: application
/json' 'https://console.senseway.net/api/login' -d '{"username":"testuser","p
assword":"badpasswd"}'

{"message":"Login Incorrect","status":401,"result":false}

#
# 正しいパスワードでログイン成功
#

$ curl -k -X POST -H 'Content-Type: application/json' -H 'Accept: application
/json' 'https://console.senseway.net/api/login' -d '{"username":"testuser","p
assword":"testpasswd"}'

{"api_token":"VeMQeaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCgcKBnBzquvvd5neAAb
P","status":200,"result":true}

#
# ユーザ情報を取得し、| で jq コマンドに渡し JSON を読みやすい形に整形
#

$ curl -k -X GET -H 'Accept: application/json' -H "Authorization: Bearer VeMQ
eaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCgcKBnBzquvvd5neAAbP" 'https://consol
e.senseway.net/api/user' | jq .

{
  "result": true,
  "status": 200,
  "user": {
    "organization_name": "センスウェイ株式会社",
    "format": 1,
    "tel": "03-1234-5678",
    "username": "staff",
    "email": "staff@senseway.net",
    "destination_name": "センスウェイ スタッフ",
    "service_plan": 1,
    "pay_status": 0,
    "zipcode": "103-0023",
    "address": "東京都",
    "address2": "中央区日本橋本町 4-8-16 千城ビル 9F"
  }
}
```

3 全 API 一覧

3.1 接続先

APIを使用するにあたり、接続先は以下のパラメータによって決定される。

プロトコル	ホスト名	ポート番号
https	console.senseway.net	443

3.2 全 API 一覧

Senseway Mission Connect API で提供している全 API は以下の通り。(2019 年 11 月現在)

リソース	機能	メソッド	URI (+パラメータ)	パラメータ (URI, Body)	レスポンス
ログイン	ログイン	POST	/api/login		ログイン情報 API トークン
ユーザ	ユーザ情報取得	GET	/api/user		
デバイス	デバイス一覧	GET	/api/device	from, limit	
	デバイス情報取得	GET	/api/device/{deveui}		
GW デバイス	GW デバイス一覧	GET	/api/gwdevice	from, limit	
	GW デバイス情報取得	GET	/api/gwdevice/{gweui}		
データ	データデバイス一覧	GET	/api/data		デバイス一覧
	データ取得	GET	/api/data/{deveui}	from, to, limit	データ
	最新データ取得	GET	/api/data/latest/{deveui}		データ
	データ送信	POST	/api/txdata/{deveui}	送信データ	

4 API リファレンス

4.1 ログイン

■機能概要

Senseway Mission Connect で取得したユーザ名、パスワードを渡して呼び出すことにより、ログインして API トークンを取得する。

取得した API トークンは他の API 使用時に Authorization ヘッダに Bearer として指定する。

API トークンには有効期間があり、期限切れになると以降の API 呼び出しで 401 エラーを返す。その場合、再ログインして API トークンを再取得する必要がある。

■リクエスト

POST /api/login

■パラメータ(JSON ボディ)

パラメータとして、{ユーザ名}と{パスワード}を以下のような JSON フォーマットのボディとして渡す。

```
{  
  "username": {ユーザ名},  
  "password": {パスワード}  
}
```

キー名	型	概要
username	string	ユーザ名
password	string	パスワード

■ステータスコード

この API は、以下のステータスコードを返す可能性がある。

コード	意味
200	成功
400	不正リクエスト (パラメータ不足等)
401	認証失敗
402	支払い要求 (ユーザ強制停止中)

■成功時レスポンス

成功時は以下のような JSON フォーマットのレスポンスを返す。

成功を示す result, status と、api_token として{APIトークン}の string が含まれる。

```
{
  "result": true,
  "status": 200,
  "api_token": {API トークン}
}
```

キー名	型	概要
result	boolean	true (成功)
status	number	ステータスコード (200 = 成功)
api_token	string	APIトークン(64バイト)

■失敗時レスポンス

失敗時は以下のような JSON フォーマットのレスポンスを返す。

失敗を示す result と、理由を示す{ステータスコード}、{エラーメッセージ}が含まれる。

```
{
  "result": false,
  "status": {ステータスコード},
  "message": {エラーメッセージ}
}
```

キー名	型	概要
result	boolean	false (失敗)
status	number	ステータスコード
message	string	エラーメッセージ

■実行例

```
#
## パスワードが異なりログイン失敗
##
$ curl -k -X POST -H 'Content-Type: application/json' -H 'Accept: application/json' 'https://console.senseway.net/api/login' -d '{"username":"testuser","password":"badpasswd"}'

{"message":"Login Incorrect","status":401,"result":false}

#
## 正しいパスワードでログイン成功
##
$ curl -k -X POST -H 'Content-Type: application/json' -H 'Accept: application/json' 'https://console.senseway.net/api/login' -d '{"username":"testuser","password":"testpasswd"}'

{"api_token":"VeMQeaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCgcKBnBzquvd5neAAbP","status":200,"result":true}
```

4.2 ユーザ情報取得

■機能概要

ログインしたユーザの情報を取得する。

■リクエスト

GET /api/user

■パラメータ

なし

■ステータスコード

この API は、以下のステータスコードを返す可能性がある。

コード	意味
200	成功
401	認証失敗 (API トークン期限切れ等)
402	支払い要求 (ユーザ強制停止中)

■成功時レスポンス

成功時は以下のような JSON フォーマットのレスポンスを返す。

成功を示す result, status と、user として{ユーザ情報}が含まれる。

```
{
  "result": true,
  "status": 200,
  "user": {
    "username": {ユーザ名},
    "email": {メールアドレス},
    "service_plan": {サービスプラン},
    "destination_name": {氏名},
    "format": {所属フラグ},
    "organization_name": {組織名},
    "tel": {電話番号},
    "zipcode": {郵便番号},
    "address": {都道府県},
    "address2": {都道府県以降住所}
  }
}
```

キー名	型	概要
result	boolean	true (成功)
status	number	ステータスコード (200 = 成功)
user	ユーザ情報	
username	string	ユーザ名
email	string	メールアドレス
service_plan	number	サービスプラン 0 = Light, 1 = Develop, 2 = Business
pay_status	number	支払い状況 0 = 正常, 1 = エラーあり, 2 = 強制停止中
destination_name	string	氏名
format	number	所属フラグ 1 = 法人, 2 = 個人
organization_name	string	組織名
tel	string	電話番号
zipcode	string	郵便番号
address	string	都道府県
address2	string	都道府県以降住所

■失敗時レスポンス

失敗時は以下のような JSON フォーマットのレスポンスを返す。

失敗を示す result と、理由を示す {ステータスコード}、{エラーメッセージ} が含まれる。

```
{
  "result": false,
  "status": {ステータスコード},
  "message": {エラーメッセージ}
}
```

キー名	型	概要
result	boolean	false (失敗)
status	number	ステータスコード
message	string	エラーメッセージ

■実行例

```
#
# API の呼び出し結果をそのまま表示
#

$ curl -k -X GET -H 'Accept: application/json' -H "Authorization: Bearer VeMQ
eaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCgcKBnBzqnuvd5neAAbP" 'https://consol
e.senseway.net/api/user'

{"user":{"username":"staff","email":"staff@senseway.net","destination_name":"
¥u30bb¥u30f3¥u30b9¥u30a6¥u30a7¥u30a4 ¥u30b9¥u30bf¥u30c3¥u30d5","service_plan
":"Develop","pay_status":"¥u6b63¥u5e38","zipcode":"103-0023","address":"¥u677
1¥u4eac¥u90fd","address2":"¥u4e2d¥u592e¥u533a¥u65e5¥u672c¥u6a4b¥u672c¥u753a4-
8-16 ¥u5343¥u57ce¥u30d3¥u30eb9F","tel":"03-1234-5678","format":"¥u6cd5¥u4eba
","organization_name":"¥u30bb¥u30f3¥u30b9¥u30a6¥u30a7¥u30a4¥u682a¥u5f0f¥u4f1a
¥u793e"},"status":200,"result":true}

#
# | で jq コマンドに渡し JSON を読みやすい形に整形した場合
#

$ curl -k -X GET -H 'Accept: application/json' -H "Authorization: Bearer VeMQ
eaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCgcKBnBzqnuvd5neAAbP" 'https://consol
e.senseway.net/api/user' | jq .

{
  "result": true,
  "status": 200,
  "user": {
    "organization_name": "センスウェイ株式会社",
    "format": 1,
    "tel": "03-1234-5678",
    "username": "staff",
    "email": "staff@senseway.net",
    "destination_name": "センスウェイ スタッフ",
    "service_plan": 1,
    "pay_status": 0,
    "zipcode": "103-0023",
    "address": "東京都",
    "address2": "中央区日本橋本町 4-8-16 千城ビル 9F"
  }
}
```

4.3 デバイス一覧

■機能概要

ログインしたユーザに紐付いたデバイス一覧を取得する。

クエリパラメータを省略した場合は一覧の数を返し、指定した場合はそれに従ってデバイス一覧を返す。

■クエスト

GET /api/device?from={開始位置}&limit={取得件数}

■パラメータ(URL クエリ)

パラメータとして URL のクエリ string に、from と limit を以下のように指定する。

from と limit は省略可能で、片側を省略した場合はそれぞれ先頭と末尾を意味するが、両方を省略した場合は、デバイス一覧は返さずに数のみを返す。

from が存在するデータよりも後の場合は空リストを返し、limit が存在するデータよりも多くなる場合は取得できる分のみを返す。

一度の呼び出しで一覧全体を取得する場合、from を省略して limit=0 を指定すればよい。

キー名	省略	型	値
from	可	number	開始位置 省略時, 0 は先頭、負数は末尾からの指定数前
limit	可	number	取得件数 省略時, 0 は末尾、負数は末尾からの指定数前の位置指定

■ステータスコード

この API は、以下のステータスコードを返す可能性がある。

コード	意味
200	成功
400	不正リクエスト (パラメータエラー等)
401	認証失敗 (API トークン期限切れ等)
402	支払い要求 (ユーザ強制停止中)
404	デバイスが見つからない

■成功時レスポンス(from, limit 省略時)

成功時は以下のような JSON フォーマットのレスポンスを返す。

成功を示す result, status と、count として{対象デバイス数}が含まれる。

```
{
  "result": true,
  "status": 200,
  "count": {対象デバイス数}
}
```

キー名	型	概要
result	boolean	true (成功)
status	number	ステータスコード (200 = 成功)
count	number	対象デバイス数 (デバイス一覧の配列数)

■成功時レスポンス(from, limit 指定時)

成功時は以下のような JSON フォーマットのレスポンスを返す。

成功を示す result, status と、count として{取得デバイス数}、devices としてデバイス情報(一覧用に簡易化したもの)の配列が含まれる。

```
{
  "result": true,
  "status": 200,
  "count": {取得デバイス数},
  "devices": [
    {
      "deveui": {DevEUI},
      "name": {名前},
      "status": {ステータス},
      "type": {デバイスタイプ}
    },
    {
      "deveui": {DevEUI},
      "name": {名前},
      "status": {ステータス},
      "type": {デバイスタイプ}
    },
    :
    (繰り返し)
  ]
}
```

キー名	型	概要
result	boolean	true (成功)
status	number	ステータスコード (200 = 成功)
count	number	取得デバイス数 (デバイス一覧の配列数)
devices	デバイス一覧情報 (一覧用に簡易化したもの) (配列)	
deveui	string	DevEUI
name	string	名前 (ユーザ側でつけたもの)
status	number	ステータス 0 = 未使用, 1 = 使用中, 2 = 停止中, 3 = 強制停止中
type	string	デバイスタイプ

■失敗時レスポンス

失敗時は以下のような JSON フォーマットのレスポンスを返す。

失敗を示す result と、理由を示す {ステータスコード}、{エラーメッセージ} が含まれる。場合によっては details として {詳細エラーメッセージ} の配列が含まれることがある。

```
{
  "result": false,
  "status": {ステータスコード},
  "message": {エラーメッセージ}
  "details": [
    {詳細エラーメッセージ},
    .
    (繰り返し)
  ]
}
```

キー名	型	概要
result	boolean	false (失敗)
status	number	ステータスコード
message	string	エラーメッセージ
details (ある場合)	string (配列)	詳細エラーメッセージ

■実行例

```
#
# クエリパラメータを省略した場合
#
$ curl -k -X GET -H 'Accept: application/json' -H "Authorization: Bearer VeMQ
eaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCgcKBnBzquvd5neAAbP" 'https://consol
e.senseway.net/api/device'

{"count":4,"status":200,"results":true}

#
# 先頭から2個のデバイス一覧を取得
#
$ curl -k -X GET -H 'Accept: application/json' -H "Authorization: Bearer VeMQ
eaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCgcKBnBzquvd5neAAbP" 'https://consol
e.senseway.net/api/device?from=0&limit=2' | jq .

{
  "result": true,
  "status": 200,
  "count": 2,
  "devices": [
    {
      "type": "ADB922S",
      "status": 1,
      "name": "Office Door Sensor 1",
      "deveui": "0000624266e31024"
    },
    {
      "type": "ADB922S",
      "status": 2,
      "name": "Meeting Room A Temperature Sensor",
      "deveui": "00008032b2a22000"
    }
  ]
}

#
# 開始位置が全体数を超えている場合、空リストしか取得できない
#
$ curl -k -X GET -H 'Accept: application/json' -H "Authorization: Bearer VeMQ
eaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCgcKBnBzquvd5neAAbP" 'https://consol
e.senseway.net/api/device?from=5' | jq .

{
  "result": true,
  "status": 200,
  "count": 0,
  "devices": []
}
```

4.4 デバイス情報取得

■機能概要

ログインしたユーザに紐付いた指定したデバイス情報を取得する。

■リクエスト

GET /api/device/{DevEUI}

■パラメータ(URLパス)

URLパスのセグメントに、パラメータとして DevEUI を string(英小文字 16 進数 16 桁)で指定する。

存在しなかったり所有デバイスでない場合はエラーを返す。

セグメント	型	概要
{DevEUI}	string	デバイスの DevEUI (英小文字 16 進数 16 桁)

■ステータスコード

この API は、以下のステータスコードを返す可能性がある。

コード	意味
200	成功
400	不正リクエスト
401	認証失敗 (API トークン期限切れ等)
402	支払い要求 (ユーザ強制停止中)
404	指定デバイスが見つからない

■成功時レスポンス

成功時は以下のような JSON フォーマットのレスポンスを返す。

成功を示す result, status と、device としてデバイス情報が含まれる。

```

{
  "result": true,
  "status": 200,
  "device": {
    "deveui": {DevEUI},
    "appkey": {AppKey},
    "appeui": {AppEUI},
    "type": {デバイスタイプ},
    "license": {ライセンス},
    "status": {ステータス},
    "use_rest_api": {REST API 使用フラグ},
    "name": {名前},
    "description": {備考 1},
    "description2": {備考 2}
  }
}

```

キー名	型	概要
result	boolean	true (成功)
status	number	ステータスコード (200 = 成功)
device	デバイス情報	
deveui	string	DevEUI
appkey	string	AppKey
appeui	string	AppEUI
type	string	デバイスタイプ
license	number	ライセンス 0 = 通常, 1 = 開発
status	number	ステータス 0 = 未使用, 1 = 使用中, 2 = 停止中, 3 = 強制停止中
use_reset_api	number	REST API 使用フラグ 0 = 使用しない, 1 = 使用する
name	string	名前 (ユーザ側でつけたもの)
description	string	備考 1 (ユーザ側でつけたもの)
description2	string	備考 2 (ユーザ側でつけたもの)

■失敗時レスポンス

失敗時は以下のような JSON フォーマットのレスポンスを返す。

失敗を示す result と、理由を示す {ステータスコード}、{エラーメッセージ} が含まれる。場合によっては details として {詳細エラーメッセージ} の配列が含まれることがある。

```
{
  "result": false,
  "status": {ステータスコード},
  "message": {エラーメッセージ},
  "details": [
    {詳細エラーメッセージ},
    .
    (繰り返し)
  ]
}
```

キー名	型	概要
result	boolean	false (失敗)
status	number	ステータスコード
message	string	エラーメッセージ
details (ある場合)	string (配列)	詳細エラーメッセージ

■実行例

```
#
# DevEUI として 16 進数 16 桁というフォーマットを満たしていない
#
$ curl -k -X GET -H 'Accept: application/json' -H "Authorization: Bearer VeM
QeaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCgcKBnBzquvd5neAAbP" 'https://cons
ole.senseway.net/api/device/00006242' | jq .
{
  "result": false,
  "status": 400,
  "message": "Parameter Error",
  "details": [
    "devEUI incorrect Format"
  ]
}

#
# 正しく DevEUI を指定した場合
#
$ curl -k -X GET -H 'Accept: application/json' -H "Authorization: Bearer VeM
QeaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCgcKBnBzquvd5neAAbP" 'https://cons
ole.senseway.net/api/device/0000624266e31024' | jq .
{
  "result": true,
  "status": 200,
  "device": {
    "description2": null,
    "deveui": "0000624266e31024",
    "appkey": "36b93f8df8cfa74cec52d4055d385ea9",
    "appeui": "abcdef0123456789",
    "type": "ADB922S",
    "license": 0,
    "status": 1,
    "use_reset_api": true,
    "name": "Office Door Sensor 1",
    "description": "オフィス入りロドア開閉センサー"
  }
}
```

4.5 GW デバイス一覧;

■機能概要

ログインしたユーザに紐付いた GW デバイス一覧を取得する。
クエリパラメータを省略した場合は一覧の数を返し、指定した場合はそれに従って GW デバイス一覧を返す。

■リクエスト

GET /api/gwdevice?from={開始位置}&limit={取得件数}

■パラメータ(URL クエリ)

URL のクエリ string に、パラメータとして from と limit を number で指定する。

from と limit は省略可能で、片側を省略した場合はそれぞれ先頭と末尾を意味するが、両方を省略した場合は、デバイス一覧は返さずに数のみを返す。

from が存在するデータよりも後の場合は空リストを返し、limit が存在するデータよりも多くなる場合は取得できる分のみを返す。

一度の呼び出しで一覧全体を取得する場合、from を省略して limit=0 を指定すればよい。

キー名	省略	型	値
from	可	number	開始位置 省略時, 0 は先頭、負数は行末からの指定数前
limit	可	number	取得件数 省略時, 0 は末尾、負数は行末からの指定数前の位置指定

■ステータスコード

この API は、以下のステータスコードを返す可能性がある。

コード	意味
200	成功
400	不正リクエスト (パラメータエラー等)
401	認証失敗 (API トークン期限切れ等)
402	支払い要求 (ユーザ強制停止中)
404	GW デバイスが見つからない

■成功時レスポンス(from, limit 省略時)

成功時は以下のような JSON フォーマットのレスポンスを返す。

成功を示す result, status と、count として{対象 GW デバイス数}が含まれる。

```
{
  "result": true,
  "status": 200,
  "count": {対象 GW デバイス数}
}
```

キー名	型	概要
result	boolean	true (成功)
status	number	ステータスコード (200 = 成功)
count	number	対象 GW デバイス数 (デバイス一覧の配列数)

■成功時レスポンス(from, limit 指定時)

成功時は以下のような JSON フォーマットのレスポンスを返す。

成功を示す result, status と、count として{取得 GW デバイス数}、gwdevices として GW デバイス情報(一覧用に簡易化したもの)の配列が含まれる。

```
{
  "result": true,
  "status": 200,
  "count": {取得 GW デバイス数}
  "gwdevices": [
    {
      "gw_eui": {GwEUI},
      "gw_eui_2": {GwEUI2},
      "status": {ステータス},
      "type": {GW デバイスタイプ}
    },
    {
      "gw_eui": {gwEUI},
      "gw_eui_2": {gwEUI2},
      "status": {ステータス},
      "type": {GW デバイスタイプ}
    },
    :
    (繰り返し)
  ]
}
```

キー名	型	概要
result	boolean	true (成功)
status	number	ステータスコード (200 = 成功)
count	number	取得 GW デバイス数 (GW デバイス一覧の配列数)
gwdevices	GW デバイス一覧情報 (一覧用に簡易化したもの) (配列)	
gw_eui	string	GwEUI
gw_eui_2	string	GwEUI2 (ない場合は null)
status	number	ステータス 0 = 未使用, 1 = 使用中, 2 = 停止中, 3 = 強制停止中
type	string	GW デバイスタイプ

■失敗時レスポンス

失敗時は以下のような JSON フォーマットのレスポンスを返す。

失敗を示す result と、理由を示す{ステータスコード}、{エラーメッセージ}が含まれる。場合によっては details として{詳細エラーメッセージ}の配列が含まれることがある。

```
{
  "result": false,
  "status": {ステータスコード},
  "message": {エラーメッセージ},
  "details": [
    {詳細エラーメッセージ},
    :
    (繰り返し)
  ]
}
```

キー名	型	概要
result	boolean	false (失敗)
status	number	ステータスコード
message	string	エラーメッセージ
details (ある場合)	string (配列)	詳細エラーメッセージ

■実行例

```
#
# クエリパラメータを省略した場合
#
$ curl -k -X GET -H 'Accept: application/json' -H "Authorization: Bearer VeMQ
eaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCgckBnBzqnuvd5neAAbP" 'https://consol
e.senseway.net/api/gwdevice'

{"count":5,"status":200,"results":true}

#
# 先頭から3個のGWデバイス一覧を取得
#
$ curl -k -X GET -H 'Accept: application/json' -H "Authorization: Bearer VeMQ
eaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCgckBnBzqnuvd5neAAbP" 'https://consol
e.senseway.net/api/gwdevice?from=0&limit=3' | jq .

{
  "result": true,
  "status": 200,
  "count": 3,
  "devices": [
    {
      "type": "GBOX",
      "status": 1,
      "gw_eui_2": null,
      "gw_eui": "00004a50d79ef211"
    },
    {
      "type": "GBOX",
      "status": 1,
      "gw_eui_2": null,
      "gw_eui": "0000911ab3d9ad7d"
    },
    {
      "type": "G-ODU",
      "status": 1,
      "gw_eui_2": "0000c789dfe25b73",
      "gw_eui": "00000b278ad9f712"
    }
  ]
}
```

4.6 GW デバイス情報取得

■機能概要

ログインしたユーザに紐付いた指定した GW デバイス情報を取得する。

■リクエスト

GET /api/gwdevice/{GwEUI}

■パラメータ(URL パス)

URL パスのセグメントに、パラメータとして GwEUI を string(英小文字 16 進数 16 桁)で指定する。
存在しなかったり所有デバイスでない場合はエラーを返す。

セグメント	型	概要
{GwEUI}	string	GW デバイスの GwEUI (英小文字 16 進数 16 桁)

■ステータスコード

この API は、以下のステータスコードを返す可能性がある。

コード	意味
200	成功
400	不正リクエスト (パラメータエラー等)
401	認証失敗 (API トークン期限切れ等)
402	支払い要求 (ユーザ強制停止中)
404	指定 GW デバイスが見つからない

■成功時レスポンス

成功時は以下のような JSON フォーマットのレスポンスを返す。

成功を示す result, status と、gwdevice として{GW デバイス情報}が含まれる。

```
{
  "result": true,
  "status": 200,
  "gwdevice": {
    "gw_eui": {GwEUI},
    "gw_eui_2": {GwEUI2},
    "status": {ステータス},
    "type": {GW デバイスタイプ},
    "description": {備考},
    "canceled": {解約申請フラグ},
    "host_destination_name": {設置先氏名},
    "host_zipcode": {設置先郵便番号},
    "host_address": {設置先都道府県},
    "host_address2": {設置先都道府県以降住所},
    "host_tel": {設置先電話番号},
    "host_format": {設置先法人フラグ},
    "host_organization_name": {設置先組織名}
  }
}
```

キー名	型	概要
result	boolean	true (成功)
status	number	ステータスコード (200 = 成功)
gwdevice	GW デバイス情報	
gw_eui	string	GwEUI
gw_eui_2	string	GwEUI2 (ない場合はnull)
status	number	ステータス 0 = 未使用, 1 = 使用中, 2 = 停止中, 3 = 強制停止中
type	string	GW デバイスタイプ
description	string	備考 (ユーザ側でつけたもの)
canceled	number	解約申請フラグ 0 = なし, 1 = 解約申請中
host_destination_name	string	設置先住所 氏名
host_zipcode	string	設置先住所 郵便番号
host_address	string	設置先住所 都道府県
host_address2	string	設置先住所 都道府県以降住所
host_tel	string	設置先住所 電話番号
host_format	number	設置先住所 法人フラグ 1 = 法人, 2 = 個人
host_organization_name	string	設置先住所 組織名

■失敗時レスポンス

失敗時は以下のような JSON フォーマットのレスポンスを返す。

失敗を示す result と、理由を示す {ステータスコード}、{エラーメッセージ} が含まれる。場合によっては details として {詳細エラーメッセージ} の配列が含まれることがある。

```
{
  "result": false,
  "status": {ステータスコード},
  "message": {エラーメッセージ},
  "details": [
    {詳細エラーメッセージ},
    :
    (繰り返し)
  ]
}
```

キー名	型	概要
result	boolean	false (失敗)
status	number	ステータスコード
message	string	エラーメッセージ
details (ある場合)	string (配列)	詳細エラーメッセージ

■実行例

```
#
# GwEUI を正しく指定して GW デバイス情報を取得
#
$ curl -k -X GET -H 'Accept: application/json' -H "Authorization: Bearer VeMQ
eaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCgckBnBzqnuvd5neAAbP" 'https://console.senseway.net/api/gwdevice/00004a50d79ef211' | jq .
{
  "result": true,
  "status": 200,
  "gwdevice": {
    "host_organization_name": "センスウェイ株式会社",
    "host_format": 1,
    "host_tel": "03-1234-5678",
    "host_address2": "中央区日本橋本町 4-8-16 千城ビル 9F",
    "host_address": "東京都",
    "gw_eui": "00004a50d79ef211",
    "gw_eui_2": null,
    "status": 1,
    "canceled": 0,
    "type": "GBOX",
    "description": "新日本橋",
    "host_destination_name": "センスウェイ スタッフ",
    "host_zipcode": "103-0023"
  }
}
```

4.7 データデバイス一覧

■機能概要

ログインしたユーザに紐付いたデバイスで送信された、取得可能なデータが存在するデバイスの一覧を取得する。

取得可能なデータは API 呼び出し時刻より 24 時間前までのデータで、それより前のデータはサーバ上から削除され、その送信デバイスも一覧には含まれない。

■リクエスト

GET /api/data

■パラメータ

なし

■ステータスコード

この API は、以下のステータスコードを返す可能性がある。

コード	意味
200	成功
401	認証失敗 (API トークン期限切れ等)
402	支払い要求 (ユーザ強制停止中)

■成功時レスポンス

成功時は以下のような JSON フォーマットのレスポンスを返す。

成功を示す result, status と、devices として {DevEUI} の配列が含まれる。

```
{
  "result": true,
  "status": 200,
  "devices": [
    {DevEUI},
    {DevEUI},
    :
    (繰り返し)
  ]
}
```

キー名	型	概要
result	boolean	true (成功)
status	number	ステータスコード (200 = 成功)
devices	デバイス一覧 (配列)	
	{DevEUI}	string DevEUI

■失敗時レスポンス

失敗時は以下のような JSON フォーマットのレスポンスを返す。

失敗を示す result と、理由を示す {ステータスコード}、 {エラーメッセージ} が含まれる。

```
{
  "result": false,
  "status": {ステータスコード},
  "message": {エラーメッセージ}
}
```

キー名	型	概要
result	boolean	false (失敗)
status	number	ステータスコード
message	string	エラーメッセージ

■実行例

```
#
# データ送信されたデバイスの一覧を取得する
#
$ curl -k -X GET -H 'Accept: application/json' -H "Authorization: Bearer VeMQ
eaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCgcKBnBzquvud5neAAbP" 'https://consol
e.senseway.net/api/data' | jq .
{
  "result": true,
  "status": 200,
  "devices": [
    "0000624266e31024",
    "00008032b2a22000"
  ]
}
```

4.8 データ取得

■機能概要

ログインしたユーザに紐付いたデバイスで、「REST API」を「使用する」に設定している状態で送信されたデータを取得する。

クエリパラメータで、取得したいデータの（弊社サーバ処理時刻での）開始時刻、終了時刻、取得件数を指定する。

クエリパラメータを省略した場合、取得可能なデータの先頭と末尾の時刻とデータ数のみを返す。

取得可能なデータは API 呼び出し時刻より 24 時間前までのデータで、それより前のデータはサーバ上から削除され、取得することはできない。

■使用法

アプリケーションを開発する場合、初回実行時等はクエリパラメータなしで API を呼び出し、取得可能なデータの時刻情報を取得した後、クエリパラメータを指定して再度 API を呼び出してデータを取得するのが望ましい。

また、一度に大量のデータを取得しようとする、HTTP のセッション時間等の制限により途中で切断される可能性があるため、大量データを取得する場合は取得件数を指定しての複数回呼び出してデータを取得するのが望ましい。

その場合、データ取得時のレスポンスの末尾データ UNIX 時刻を利用して、次に取得する際の開始時刻パラメータを決めるとよい。

(LoRaWAN の仕様上、1 つのデバイスですぐにデータが送信されることはあり得ないので、+1 秒後等の数字を指定すれば取得漏れは起こらない。)

■リクエスト

GET /api/data/{DevEUI}?from={開始時刻}&to={終了時刻}&limit={取得件数}

■パラメータ(URL パス)

URL パスのセグメントに、パラメータとして DevEUI を string(英小文字 16 進数 16 桁)で指定する。

指定年月日に指定した DevEUI のデータが存在しなかった場合はエラーを返す。

セグメント	型	概要
{DevEUI}	string	デバイスの DevEUI (英小文字 16 進数 16 桁)
指定した DevEUI のデータが存在しなかった場合はエラーを返す		

■パラメータ(URL クエリ)

URL のクエリ string に、パラメータとして from, to, limit を number で指定する。

from, to, limit は省略可能で、すべてを省略すると、存在するデータの先頭と末尾の UNIX 時刻と、データ数のみを返す。

データの末尾は通常 to か limit のどちらかを指定するが、両方指定した場合は先に現れる制限に従う。

いずれのパラメータも、存在するデータを超過して指定した場合、存在する範囲のデータのみを返す。

指定した範囲のデータが存在しない場合は、エラーではなく空リストを返す。

存在するデータ全体を取得する場合、from, to を省略して limit=0 のみを指定すればよい。

キー名	省略	型	値
from	可	number	開始時刻を UNIX 時刻にて指定 省略時は存在するデータの先頭を指す
to	可	number	終了時刻を UNIX 時刻にて指定 省略時は存在するデータの末尾を指す
limit	可	number	取得件数 省略時, 0 は無制限を意味する

すべてを省略すると、存在するデータの先頭と末尾の UNIX 時刻と、データ数のみを返す
データの末尾は通常 to か limit どちらかを指定するが、両方指定した場合は先に現れる制限に従う
いずれのパラメータも、存在するデータを超過して指定した場合、存在する範囲のデータのみを返す
指定した範囲のデータが存在しない場合は、エラーではなく空リストを返す
存在するデータ全体を取得する場合、from, to を省略して limit=0 のみを指定すればよい

■ステータスコード

この API は、以下のステータスコードを返す可能性がある。

コード	意味
200	成功
400	不正リクエスト (パラメータエラー等)
401	認証失敗 (API トークン期限切れ等)
402	支払い要求 (ユーザ強制停止中)
404	データが見つからない

■成功時レスポンス(from, to, limit 省略時)

成功時は以下のような JSON フォーマットのレスポンスを返す。

成功を示す result, status と、from として{先頭データの UNIX 時刻}、to として{末尾データの UNIX 時刻}、count として{対象データ数}が含まれる。

```
{
  "result": true,
  "status": 200,
  "from": {先頭データの UNIX 時刻},
  "to": {末尾データの UNIX 時刻},
  "count": {対象データ数}
}
```

キー名	型	概要
result	boolean	true (成功)
status	number	ステータスコード (200 = 成功)
from	number	先頭データの UNIX 時刻
to	number	末尾データの UNIX 時刻
count	number	対象データ数

■成功時レスポンス(from, to, limit 指定時)

成功時は以下のような JSON フォーマットのレスポンスを返す。

成功を示す result, status と、from として{先頭データの UNIX 時刻}、to として{末尾データの UNIX 時刻}、count として{取得データ数}、contents としてデータの配列が含まれる。個々のデータには time として{サーバ処理 UNIX 時刻}がつけられており、from, to 等はこれを対象に指定する。

```
{
  "result": true,
  "status": 200,
  "from": {先頭データの UNIX 時刻},
  "to": {末尾データの UNIX 時刻},
  "count": {取得データ数},
  "contents": [
    {
      "time": {サーバ処理 UNIX 時刻},
      "mod": {
        "devEUI": {DevEUI},
        "dr": {DR 値},
        "mt": {メッセージタイプ},
        "port": {ポート番号},
        "data": {データ},
        "cnt": {Up カウント},
        "fq": {送信周波数}
      },
      "gw": [
        {
          "gwid": {GwEUI},
          "snr": {SNR 値},
          "rssi": {RSSI 値},
          "date": {データ送信時刻}
        },
        ...
        (繰り返し)
      ]
    },
    ...
    (繰り返し)
  ]
}
```

キー名	型	概要
result	boolean	true (成功)
status	number	ステータスコード (200 = 成功)
from	number	先頭データ UNIX 時刻
to	number	末尾データ UNIX 時刻
count	number	取得データ数 (データコンテンツの配列数)
contents	データコンテンツ (配列)	
time	number	サーバ処理 UNIX 時刻
mod	モジュールデータ	
devEUI	string	DevEUI
dr	string	DR 値
mt	string	メッセージタイプ "ucnf" - 未確認, "cnf" - ACK 確認
port	number	ポート番号 (1 - 223)
data	string	データ 16 進数文字列
cnt	number	Up カウント
fq	number	周波数 (単位:MHz)
gw	ゲートウェイデータ (配列)	
gwid	string	GwEUI
snr	number	SNR 値
rssi	number	RSSI 値
date	string	データ送信時刻 UTC ISO8601 フォーマット

■失敗時レスポンス

失敗時は以下のような JSON フォーマットのレスポンスを返す。

失敗を示す result と、理由を示す {ステータスコード}、{エラーメッセージ} が含まれる。場合によっては details として {詳細エラーメッセージ} の配列が含まれることがある。

```
{
  "result": false,
  "status": {ステータスコード},
  "message": {エラーメッセージ},
  "details": [
    {詳細エラーメッセージ},
    .
    .
    (繰り返し)
  ]
}
```

キー名	型	概要
result	boolean	false (失敗)
status	number	ステータスコード
message	string	エラーメッセージ
details (ある場合)	string (配列)	詳細エラーメッセージ

■実行例

```
#
# デバイス 0000624266e31024 から送信されたデータの時刻と数を取得
#
$ curl -k -X GET -H 'Accept: application/json' -H "Authorization: Bearer VeMQ
eaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCgckBnBzqnuvd5neAAbP" 'http://consol
e.senseway.net/api/data/0000624266e31024' | jq .
{
  "result": true,
  "status": 200,
  "from": 1553662067,
  "to": 1553662150,
  "count": 3
}

#
# デバイス 0000624266e31024 から送信されたデータを 2 件取得
#
$ curl -k -X GET -H 'Accept: application/json' -H "Authorization: Bearer VeMQ
eaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCgckBnBzqnuvd5neAAbP" 'http://consol
e.senseway.net/api/data/0000624266e31024?from=1553662067&limit=2' | jq .
{
  "result": true,
  "status": 200,
  "from": 1553662067,
  "to": 1553662108,
  "count": 2,
  "contents": [
    {
      "time": 1553662067,
      "mod": {
        "port": 3,
        "dr": "2",
        "devEUI": "0000624266e31024",
        "mt": "ucnf",
        "data": "0e02f30000",
        "cnt": 1,
        "fq": 923.2
      }
    },
    {
      "gw": [
        {
          "gwid": "00004a50d79ef211",
          "snr": 11,
          "rssi": -57,

```

```

        "date": "2019-03-27T04:47:45.738043Z"
      },
      {
        "gwid": "0000911ab3d9ad7d",
        "snr": 10.8,
        "rssi": -66,
        "date": "2019-03-27T04:47:45.743951Z"
      }
    ]
  },
  {
    "time": 1553662108,
    "mod": {
      "port": 3,
      "dr": "2",
      "devEUI": "0000624266e31024",
      "mt": "ucnf",
      "data": "0e02f40000",
      "cnt": 2,
      "fq": 927.8
    },
    "gw": [
      {
        "gwid": "00004a50d79ef211",
        "snr": 10.8,
        "rssi": -54,
        "date": "2019-03-27T04:48:26.660421Z"
      },
      {
        "gwid": "0000911ab3d9ad7d",
        "snr": 11.6,
        "rssi": -68,
        "date": "2019-03-27T04:48:26.671209Z"
      }
    ]
  }
]
}

#
# 上記の結果の"to"を利用して次のデータを2件取得（実際は1件しかなかった）
#

$ curl -k -X GET --header 'Accept: application/json' --header "Authorization: Bearer VeMQeaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCgcKBnBzqnuvd5neAAbP" 'http://console.senseway.net/api/data/0000624266e31024?from=1553662109&limit=2' | jq .

{
  "result": true,
  "status": 200,
  "from": 1553662150,
  "to": 1553662150,
  "count": 1,
  "contents": [
    {
      "time": 1553662150,

```

```
"mod": {
  "port": 3,
  "dr": "2",
  "devEUI": "0000624266e31024",
  "mt": "ucnf",
  "data": "0e02f50000",
  "cnt": 3,
  "fq": 923.4
},
"gw": [
  {
    "gwid": "00004a50d79ef211",
    "snr": 13,
    "rssi": -87,
    "date": "2019-03-27T04:49:10.210874Z"
  },
  {
    "gwid": "0000911ab3d9ad7d",
    "snr": 9.2,
    "rssi": -72,
    "date": "2019-03-27T04:49:10.237645Z"
  }
]
]
```

4.9 最新データ取得

■機能概要

ログインしたユーザに紐付いたデバイスで、「REST API」を「使用する」に設定している状態で送信された最新のデータを取得する。

取得可能なデータはAPI呼び出し時刻より24時間前までのデータで、それより前のデータはサーバ上から削除され、以後取得不可能になる。

■リクエスト

GET /api/data/latest/{DevEUI}

■パラメータ(URLパス)

URLパスのセグメントに、パラメータとしてDevEUIをstring(英小文字16進数16桁)で指定する。

指定したDevEUIのデータが存在しなかった場合はエラーを返す。

セグメント	型	概要
{DevEUI}	string	デバイスのDevEUI (英小文字16進数16桁)
指定したDevEUIのデータが存在しなかった場合はエラーを返す		

■ステータスコード

このAPIは、以下のステータスコードを返す可能性がある。

コード	意味
200	成功
400	不正リクエスト (パラメータエラー等)
401	認証失敗 (APIトークン期限切れ等)
402	支払い要求 (ユーザ強制停止中)
404	データが見つからない

■成功時レスポンス

成功時は以下のような JSON フォーマットのレスポンスを返す。

成功を示す result, status と、count として{取得データ数} = 1、contents としてデータの配列(要素数 1)が含まれる。

```
{
  "result": true,
  "status": 200,
  "from": {取得データ UNIX 時刻},
  "to": {取得データ UNIX 時刻},
  "count": 1,
  "contents": [
    {
      "time": {サーバ処理 UNIX 時刻},
      "mod": {
        "devEUI": {DevEUI},
        "dr": {DR 値},
        "mt": {メッセージタイプ},
        "port": {ポート番号},
        "data": {データ},
        "cnt": {Up カウント},
        "fq": {送信周波数}
      },
      "gw": [
        {
          "gwid": {GwEUI},
          "snr": {SNR 値},
          "rssi": {RSSI 値},
          "date": {データ送信時刻}
        },
        :
        (繰り返し)
      ]
    }
  ]
}
```

キー名	型	概要
result	boolean	true (成功)
status	number	ステータスコード (200 = 成功)
from	number	取得データ UNIX 時刻
to	number	取得データ UNIX 時刻
count	number	取得データ数 = 1 (データコンテンツの配列要素数)
contents	データコンテンツ (要素数 1 の配列)	
time	number	サーバ処理 UNIX 時刻
mod	モジュールデータ	
devEUI	string	DevEUI
dr	string	DR 値
mt	string	メッセージタイプ "ucnf" - 未確認, "cnf" - ACK 確認
port	number	ポート番号 (1 - 223)
data	string	データ 16 進数文字列
cnt	number	Up カウント
fq	number	周波数 (単位:MHz)
gw	ゲートウェイデータ (配列)	
gwid	string	GwEUI
snr	number	SNR 値
rssi	number	RSSI 値
date	string	データ送信時刻 UTC ISO8601 フォーマット

■失敗時レスポンス

失敗時は以下のような JSON フォーマットのレスポンスを返す。

失敗を示す result と、理由を示す {ステータスコード}、{エラーメッセージ} が含まれる。場合によっては details として {詳細エラーメッセージ} の配列が含まれることがある。

```
{
  "result": false,
  "status": {ステータスコード},
  "message": {エラーメッセージ},
  "details": [
    {詳細エラーメッセージ},
    .
    (繰り返し)
  ]
}
```

キー名	型	概要
result	boolean	false (失敗)
status	number	ステータスコード
message	string	エラーメッセージ
details (ある場合)	string (配列)	詳細エラーメッセージ

■実行例

```
#
# デバイス 0000624266e31024 から送信された
# 最新のデータを取得
#
$ curl -k -X GET -H 'Accept: application/json' -H "Authorization: Bearer VeMQ
eaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCgcKBnBzqnuvd5neAAbP" 'https://consol
e.senseway.net/api/data/latest/0000624266e31024' | jq .
{
  "result": true,
  "status": 200,
  "from": ,
  "to": ,
  "count": 1,
  "contents": [
    {
      "time": ,
      "mod": {
        "port": 3,
        "dr": "2",
        "devEUI": "0000624266e31024",
        "mt": "ucnf",
        "data": "0e02f30000",
        "cnt": 2,
        "fq": 927.8
      }
    },
    {
      "gw": [
        {
          "gwid": "00004a50d79ef211",
          "snr": 11.2,
          "rssi": -58,
          "date": "2019-03-27T06:23:39.738043Z"
        },
        {
          "gwid": "0000911ab3d9ad7d",
          "snr": 10.6,
          "rssi": -63,
          "date": "2019-03-27T06:23:39.743951Z"
        }
      ]
    }
  ]
}
```

4.10 データ送信

■機能概要

ログインしたユーザに紐付いたデバイスにデータを送信する。

LoRaWAN の ClassA では、デバイスへ送信する DownLink データは、いったんサーバに保持され、デバイスからの次の UpLink データの送信があったタイミングで、デバイスへ送信される。

デバイス毎に保持する送信キューは 1 つで、まだデバイスがデータを受信していない状態で再度データ送信を行うと、キューは上書きされる。(ただし、データ送信時にデバイスに ACK を要求した場合、デバイスからの該当 ACK を受け取るまで上書きはできない。)

デバイスが受信できるデータ長等は、LoRaWAN AS923 とデバイスの設定に依存する。

※ACK については、MQTT での受信はできるが、本 REST API では受信できない。(2019 年 10 月現在)

■リクエスト

POST /api/txdata/{DevEUI}

■パラメータ(URL パス)

URL パスのセグメントに、パラメータとして DevEUI を string で指定する。

指定した DevEUI のデータが存在しなかった場合はエラーを返す。

セグメント	型	概要
{DevEUI}	string	デバイスの DevEUI (英小文字 16 進数 16 桁)
指定 DevEUI のデータが存在しない場合はエラーを返す		

■パラメータ(JSON ボディ)

パラメータとして、{ユーザ名}&{パスワード}を以下のような JSON フォーマットのボディとして渡す。

```
{
  "data": {データ},
  "port": {ポート番号},
  "cnf": {ACK 要求フラグ},
  "ref": {ACK 識別用 string}
}
```

キー名	型	省略	概要
data	string	不可	データ 16 進数文字列
port	number	不可	ポート番号 (1 - 223)
cnf	boolean	可	デバイスへの ACK 要求フラグ 要求する - true, 要求しない - false (デフォルト)
ref	string	可	ACK を要求する場合の識別用 string

■ステータスコード

この API は、以下のステータスコードを返す可能性がある。

コード	意味
200	成功
400	不正リクエスト (パラメータエラー等)
401	認証失敗 (APIトークン期限切れ等)
402	支払い要求 (ユーザ強制停止中)
500	サーバエラー

■成功時レスポンス

成功時は以下のような JSON フォーマットのレスポンスを返す。

成功を示す result, status が含まれる。

```
{
  "result": true,
  "status": 200
}
```

キー名	型	概要
result	boolean	true (成功)
status	number	ステータスコード (200 = 成功)

■失敗時レスポンス

失敗時は以下のような JSON フォーマットのレスポンスを返す。

失敗を示す result と、理由を示す {ステータスコード}、{エラーメッセージ} が含まれる。場合によっては details として {詳細エラーメッセージ} の配列が含まれることがある。

```
{
  "result": false,
  "status": {ステータスコード},
  "message": {エラーメッセージ},
  "details": [
    {詳細エラーメッセージ},
    :
    (繰り返し)
  ]
}
```

キー名	型	概要
result	boolean	false (失敗)
status	number	ステータスコード
message	string	エラーメッセージ
details (ある場合)	string (配列)	詳細エラーメッセージ

■実行例

```
#
# DevEUI として 16 進数 16 桁というフォーマットを満たしていない
#

$ curl -k -X POST -H 'Content-Type: application/json' -H 'Accept: application/
json' -H "Authorization: Bearer VeMQeaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCg
cKBnBzqnuvd5neAAbP" 'https://console.senseway.net/api/txdata/00006242' -d '{"p
ort":7,"data":"61626364","cnf":false,"ref":"abcd0123"}' | jq .

{
  "result": false,
  "status": 400,
  "message": "Parameter Error",
  "details": [
    "devEUI incorrect Format"
  ]
}

#
# port が number でなく string で、cnf が論理二値でなく string になっている
#

$ curl -k -X POST -H 'Content-Type: application/json' -H 'Accept: application/
json' -H "Authorization: Bearer VeMQeaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCg
cKBnBzqnuvd5neAAbP" 'https://console.senseway.net/api/txdata/0000624266e31024'
-d '{"port":"7","data":"61626364","cnf":"true","ref":"abcd0123"}' | jq .

{
  "result": false,
  "status": 400,
  "message": "Parameter Error",
  "details": [
    "port not Integer",
    "cnf not boolean"
  ]
}

#
# 指定デバイス 0000624266e31024 へデータを送信
#

$ curl -k -X POST -H 'Content-Type: application/json' -H 'Accept: application/
json' -H "Authorization: Bearer VeMQeaBV7AA67ND44GuSWLgNC7zRbhnBQGlcj3S3bLeZCg
cKBnBzqnuvd5neAAbP" 'https://console.senseway.net/api/txdata/0000624266e31024'
-d '{"port":7,"data":"61626364","cnf":false,"ref":"abcd0123"}' | jq .

{
  "result": true,
  "status": 200
}
```

Senseway Misson Connect
REST API 仕様書

2019/11/15 Ver. 0.94