

facet Cloud API 仕様書 v4.0

目次

本書の概要.....	4
API の概要.....	4
クラウドサーバーに保存された画像の参照について	7
リクエストヘッダ及び認証トークンについて	10
お申込みにあたって	11
AI カメラに対応する API.....	13
機器情報の取得	14
デバイス同期要求.....	16
APB 状態の修復	17
認証時のプッシュ通知先 URL の取得	19
認証時のプッシュ通知先 URL の登録	20
認識ログの取得	21
認識ログの削除	25
操作ログの取得	27
認識の通知（プッシュ通知）	29
人物の取得.....	31
人物画像のチェック	34
クラウドサーバーに登録された人物の検索.....	36
クラウドサーバーへ人物を登録.....	40
クラウドサーバーに登録された人物データをデバイスに登録	42
クラウドサーバーに登録された人物データの削除.....	45
クラウドサーバーに登録された人物とデバイスの関連付けの削除	46
クラウドサーバーに登録された人物の通行可能時間帯の取得	47
クラウドサーバーに登録された人物へ通行可能時間帯の登録	49
デバイスに登録された人物データの検索	51
デバイスに登録された人物データをクラウドサーバーに登録	54
デバイスに登録された人物データの削除	56
デバイスに登録された全人物データの削除.....	57
クラウドサーバーに登録された人物へカード情報の登録.....	58
クラウドサーバーに登録された人物のカード情報の取得.....	60
人物の顔の撮影と撮影画像の取得	61
デバイスに登録された人物画像との類似度チェック	62
デバイスに登録された人物画像の取得.....	63
クラウドサーバーに登録された区分の取得.....	64

クラウドサーバーへ区分の登録.....	65
クラウドサーバーに登録された区分の削除.....	66
設定の取得.....	67
設定の更新.....	71
待ち受けロゴ画像の取得.....	76
待ち受けロゴ画像の設定.....	77
システム情報の取得.....	78
デバイスの再起動（即座）.....	79
デバイスのドア開錠(即座).....	80
デバイスへの休止指示.....	81
デバイスへの運用指示.....	82
デバイス状態の確認（運用/休止）.....	83
デバイス画面へのメッセージ表示.....	84
デバイスの自動起動設定を取得.....	85
デバイスの自動起動設定を更新.....	86
設定のエクスポート.....	87
設定のインポート.....	88
ファームウェアアップデート.....	89

本書の概要

本書は facetCloud ver4.0 を、外部システムから利用するための API 仕様書となります。

主な機能

facetCloud API は主に、下記の機能をサポートします。

- ・ 機器情報の取得
- ・ 認識ログの取得
- ・ 認識する人物の登録及び検索
- ・ 機器の操作/メンテナンスの一部

クラウドへ蓄積する個人情報について

facetCloud は株式会社データスコープが運営するクラウドサーバーで運用されます。

個人情報保護、肖像権の観点から「個人名をマスクして保存」「顔写真を保存したくない」などの対応が可能です。

API ご利用の際に、どの個人情報をマスクするか（保存しないようにしたいか）を営業窓口にお申し付けください。

ただし、これにより、一部の機能が制限される場合があります。

また、facetCloud をご利用頂くにあたり、個人情報/肖像権に関するあらゆるトラブルについて、株式会社データスコープは一切の責任を負いかねます。必ず、データの蓄積範囲について、明確にご判断を頂く必要があります。

セキュリティについて

facetCloud API は、全ての通信内容を SSL 暗号化します。

また、発信元 IP アドレスを特定の値にのみ、制限することも出来ます。特定のサーバーからのみ、API を利用する事が想定される場合には、可能な限り IP アドレス制限のお申し込みをお勧め致します。IP アドレス制限のお申し込みは営業窓口にお申し付けください。

API の概要

プロトコル	https
Method	GET もしくは POST ※ 機能ごとに特筆が無い限りは GET と POST の両方がサポートされます。
応答の形式	JSON(Content-type: application/json)

http レスポンスステータス

通常、API サーバーは処理結果に応じて下記表のようなレスポンスステータスを返却します。

また、リクエストパラメータ「nochange-response-status=1」を指定した場合には、API サーバーはリクエストパラメータエラーなどが発生していたとしても可能な限り「200 OK」を返却します。これは、一部のソフトウェアライブラリにおいてステータス 400 などの場合に、ボディ部を廃棄するロジックとなっており、エラーメッセージの詳細が消失してしまうケースがあるため、これを防止するための措置です。

200 OK	正常に処理が受け付けられており、ボディ部から目的の JSON データを取得する事が出来ます。 但し、200 OK であっても全てが完全に正常に処理されたとは限らず、「〇〇件まで正常に処理されたが途中でデバイスとの接続が切れたため中断した」といったケースも 200 OK で返却される場合があります。詳細は各 API 仕様をご確認下さい。
400 Bad Request	リクエストパラメータに誤りがあります。必須であるパラメータを空で送信した。数値で指定すべきパラメータを文字列で指定した。などが考えられます。 このエラーが発生した場合は API に関わらず共通して下記のような形式の JSON が返却されます。 エラーは複数行で構成される場合があります。 <pre>{ "error": true, "errors": [{"name": "from", "message": "検索日時 (from) は日付形式で指定する必要があります。"}, {"name": "name", "message": "氏名の文字数が多すぎます。100 文字以内で指定して下さい。"}] }</pre>
401 Unauthorized	IP 制限をお申込み頂いた場合、申請頂いた IP と異なる発信元からアクセスするとこのエラーが返却されます。
403 Forbidden	認証のためのトークンに誤りがあります。 ds-api-token(後述)が適切に設定されているかをご確認ください。
404 Not Found	API の URL 指定に誤りがあります。
414 URI Too Long	GET で極端に長すぎるパラメータを設定した場合に発生する場合があります。POST での送信をお試しください。

500 Internal Server Error	API サーバー側で何かしらの予期せぬエラーが発生しています。一時的な問題の可能性が高いため、少し時間を置いてから再度お試しください。
502 Bad Gateway	API サーバーが一時的なメンテナンス中です。短時間で復旧する可能性が高いため、少し時間を置いてから再度お試しください。
503 Service Unavailable	2つのケースがあります。 1) レスポンスの本文に何も JSON の値が無い場合：API サーバーが一時的なメンテナンス中です。短時間で復旧する可能性が高いため、少し時間を置いてから再度お試しください。 2) レスポンス本文に JSON 文字列が設定されている場合：デバイスとの通信が正常に行えない場合にはこのエラーが返却されます。 <pre>{ "error": true, "errors": [{"name": "device", "message": "デバイスとの通信が正常に行われませんでした。"}] }</pre> <pre>{ "error": true, "errors": [{"name": "device", "message": "デバイス [S99999] はクラウドサーバーに接続されていません。"}] }</pre> このようなエラーが見られた場合、デバイスの通信経路の確認や、電源の状態をご確認ください。また、通信経路や電源に問題が無い場合には、デバイスの再起動をお試しください。

クラウドサーバーに保存された画像の参照について

方針

人物の顔写真や、画像認識された際の顔写真など、機密性の高い画像ファイルについて本 API では様々なセキュリティ対策を講じており、強固なセキュリティと参照時の柔軟性の両立を強く意識しています。

画像ファイルの参照方法について

「認識ログの取得 (/log/getRecogLog)」機能や「クラウドサーバーに登録された人物の検索 (/person/getPersonFromCloud)」機能などにおいて、人物の写真画像を参照する場合、API のレスポンスデータに含まれる Cookie の値が画像を取得するためのキーとなります。この Cookie はワンタイムに発行されるものであり、再利用と事前推測は出来ません。また、API 実行時においては、参照元（発信元）IP アドレスや、Cookie の有効期限などをご利用状況にあわせて設定して頂く事が可能です。

Cookie の値は通常、下記のように API レスポンス時の JSON 文字列に含まれます。

```
"pictureCookie": {
  "CloudFront-Policy": "xxxxxxx",
  "CloudFront-Signature": "xxxxxxx",
  "CloudFront-Key-Pair-Id": "xxxxxxx",
}
```

画像参照に必要なものは「CloudFront-Policy」「CloudFront-Signature」「CloudFront-Key-Pair-Id」の3つの値です。

実装方式ごとの画像参照方法

【API クライアントプログラムを HTML のみ (JavaScript) で実装している場合】

jsonp による API アクセスの場合、クラウドサーバーは本文 JSON 中の pictureCookie に加えて set-cookie レスポンスヘッダを自動的に返却します。そのため、クライアントプログラム側 (JavaScript 側) では特に対処は不要です。

【API クライアントプログラムを Java や PHP などのサーバーサイドプログラムで実装しており、自システムのサーバーに**画像ファイルを保存する場合**】

通常、言語ごとに提供される HTTP クライアントライブラリには、サーバーに対するアクセスの際に Cookie の値を設定する機能がサポートされています。「CloudFront-Policy」「CloudFront-Signature」「CloudFront-Key-Pair-Id」の3つの値を設定して頂く事により、画像ファイルのバイナリをサーバー上に取得する事が出来ます。

【API クライアントプログラムを Java や PHP などのサーバーサイドプログラムで実装しており、自システムのサーバーに**画像ファイルを保存しない場合**】

お作りになっている自システムのドメインから画像のための参照 Cookie を set-cookie でブラウザに設定する事は出来ません。異なるドメインに対する set-cookie は通常、ブラウザのセキュリティ機能によりブロックされるためです。

この制限を回避するために、下記のように、<head>タグ内にクラウドサーバーが提供するツールプログラムの読み込みタグを差し込む事で対応を行って下さい。

```
<head>
...

<script src="https://【HOST】
/api1/tool/setPictureCookie.js?p=xxxxxxx&s=xxxxxxx&k=xxxxxxx"></script>

...

</head>
```

setPictureCookie.js は API サーバーが提供するツール機能です。画像を表示するための タグよりも先に読み込まれている必要があります。

この js をブラウザからアクセスする事により、画像ファイルと同一ドメインからの set-cookie を可能とします。

js のリクエストパラメータには、それぞれ API から返却された下記の値を指定して下さい。

p : CloudFront-Policy

s : CloudFront-Signature

k : CloudFront-Key-Pair-Id

下記はこのケースにおける PHP を使った最低限の実装例です。

ホスト名、API トークン、シリアル No はご契約ごとに異なりますので変更の上でお試し下さい。

```

<?php
$host      = "XXX.fc2-cloud.com";// API サーバーのホスト名
$apiToken  = "XXXXXXXXXX";      // API トークン
$serialNo  = "XXXXXXXXXX";      // シリアル No

// 認識ログを取得。
$res = file_get_contents("https://{ $host }/api1/log/getRecogLog?ds-api-
token={$apiToken}&serialNo={$serialNo}");

$data      = json_decode($res, true);// JSON デコード。
$cookie    = $data["pictureCookie"]; // 画像参照 Cookie
$list      = $data["list"];          // ログのリスト

?><html>
<head>
  <script src="https://<?php echo $host ?>/api1/tool/setPictureCookie.js?p=<?php echo
$cookie["CloudFront-Policy"] ?>&s=<?php echo $cookie["CloudFront-
Signature"] ?>&k=<?php echo $cookie["CloudFront-Key-Pair-Id"] ?>"></script>
</head>
<body>
<?php foreach ($list as $log) { ?>
  " />
<?php } ?>
</body>
</html>

```

リクエストヘッダ及び認証トークンについて

全ての API のリクエストヘッダもしくはリクエストパラメータには認証トークン「ds-api-token」を含める必要があります。

認証トークンは API ご利用お申込みの際に、お伝えを致します。

また、何かしらの事情で認証トークンが外部に漏れてしまった場合などには変更も承りますので、変更の旨をご連絡ください。

認証トークンを含んだリクエストの例は下記のようになります。

例 1) リクエストヘッダに設定する場合

```
GET /device/getDeviceList HTTP/1.1
User-Agent: xxxxxxxx
Host: xxx.xxx
ds-api-token: YdxeGjJ7pUTaHD1QmDlu4jbJId5sMgb0ANag4Ax5yVErRzQJNPiRV7oJ9pFcBaNF
```

例 2) リクエストパラメータに設定する場合

```
GET /device/getDeviceList?ds-api-token=YdxeGjJ7pUTaHD1QmDlu4jbJId5sMgb0ANag4Ax5yVErRzQJNPiRV7oJ9pFcBaNF HTTP/1.1
User-Agent: xxxxxxxx
Host: xxx.xxx
```

ds-api-token は 64 文字の英数時（大文字小文字混在）で構成されます。

User-Agent については API サーバー側では現状、読み捨てる形になっていますので、長すぎなければ任意の値で構いません。

また、「この方式では社内のセキュリティポリシーに準拠出来ない」などのご事情がございましたらお気軽にお申し付け下さい。別途お見積りの上、ポリシーに準拠出来るようなご要望に応じた認証アルゴリズムを設ける事も出来ます。

排他制御について

デバイスへの負荷防止や処理競合防止を目的として、一部の API は排他制御を持ち、同時実行を許容しません。

排他制御が有りとして定義された API において、既に他のリクエストの処理が進行中に別のリクエストを受信した場合、API サーバーは 400 Bad Request と共に下記のような結果を返却します。

```
{
  "error": true,
  "errors": [
    { "name": "exclusion", "message": "他のリクエストを処理中です。" }
  ]
}
```

お申込みにあたって

API ご利用お申込み時に頂く情報

IP 制限の有無	「有り」をご希望の場合は、ご指定頂いた IP アドレス以外からの通信を拒否します。可能な限り、「有り」でお申込み頂く事を推奨致します。
個人名の保存可否	クラウドサーバー上に、人物の氏名を保存するかどうかをご指定下さい。「保存しない」をご指定頂いた場合、個人名は一切クラウドサーバーには保存されません。 人物登録時の氏名：ご契約ごとに保存可否を設定して頂けます。 認識ログの氏名：お申込みデバイスごとに保存可否を設定して頂けます。
認識顔画像の保存可否	クラウドサーバー上に、認識ログの写真画像を保存するかどうかをご指定下さい。「保存しない」をご指定頂いた場合、認識時の顔写真は一切クラウドサーバーには保存されません。（この可否は認識ログ写真に対するものであり、人物登録時の顔写真は機能維持のために必須のため、非保存化する事は出来ません） お申込みデバイスごとに設定して頂けます。

API のご利用準備完了後にデータスコープからお知らせする情報

API 利用のための認証トークン	機密情報に該当します。流失に注意の上、大切に保存してください。もし流失してしまった場合、変更が可能です。ご連絡ください。
サーバー接続情報	デバイスの管理画面より設定して頂くクラウドサーバーのアドレス情報をお伝えいたします。

認識時のプッシュ通知元 IP アドレス	「認識時のプッシュ通知先 URL」をご利用される場合には、通知元サーバーの IP アドレスをお伝えいたします。 通知を受信するサーバーでは可能な限り、発信元 IP アドレス制限を行って下さい。
API サーバーの HOST	API サーバーにアクセスするためのホストをお伝え致します。ご契約の内容やプランによって、振り分けるサーバーが異なりますので、ご契約ごとにホストが変わります。

AI カメラに対応する API

facetCloud の ver4.0 より、下記 API を AI カメラに対応しました。

- getPerson
- toDevice
- getPersonFromDevice
- deletePersonFromDevice
- getSystemInfo
- getPersonPictureFromDevice

上記以外のデバイス側と通信を行う API を実行した際は、以下のエラーを返却し、動作保証ができかねますので実行を行わないようお願いいたします。

```
{
  "error": true,
  "errors": [
    {
      "name": "device",
      "message": "デバイス[7L11R010014]がエラーを返却しています。"
    }
  ]
}
```

デバイスと通信を行わない、クラウド内での処理（取得や削除）を行う API については、利用可能となります。

例) AI カメラのログを取得したい場合、「getRecogLog」は使用可能

※AI カメラについては、カード認証や通行可能時間帯は非対応となっております。そのため、それらが登録された人物を「toDevice」を行った場合、AI カメラ側へは反映されず、クラウド側のみに登録されている状態となります。

機器情報の取得

API URL	https://[HOST]/api1/device/getDevice
排他制御	無し

クラウド契約を頂いているデバイスの一覧が取得できます。

実際にご購入・設置してあったとしても、クラウド契約の対象外のデバイスについてはこの機能では取得されません。

リクエストパラメータ

無し

レスポンスサンプル

```
[
  {
    "serialNo": "D00001",
    "description": "1F 西側ゲート",
    "lastAccess": "2020/08/06 16:01:08",
    "lastRecog": "2020/08/06 16:00:05"
  },
  {
    "serialNo": "D00002",
    "description": "1F 東側ゲート",
    "lastAccess": "2020/08/05 11:15:00",
    "lastRecog": "2020/08/04 12:00:11"
  },
  {
    "serialNo": "D00003",
    "description": "2F エレベーター横",
    "lastAccess": "2020/08/06 16:03:57",
    "lastRecog": "2020/08/06 16:00:00"
  }
]
```

レスポンス仕様

項目	型	概要
serialNo	文字列	デバイスのシリアルN oです。
description	文字列	デバイスに対する説明文です。
lastAccess	日付時刻	最後にこのデバイスと API サーバー間で通信が行われ

		た日時です。顔が認識されていなくても死活監視のために一定時間ごとに通信が行われます。
lastRecog	日付時刻	このデバイスで最後に顔が認識された日時です。この日時はデバイス内部時計時刻のため、時刻同期が正常に行われていないデバイスの場合、実際の時刻とは若干の差異が起きる可能性があります。

デバイス同期要求

API URL	https://[HOST]/api1/device/sync
排他制御	有り（serialNo 単位に一リクエストのみを処理可能）

デバイスに即座に通信を行い、最新のログデータの取得を行います。

通常、API サーバーは、認識ログについてはデバイスからのプッシュデータを受信し、常に最新状態を保ちます。一過性の通信障害によるプッシュの「取りこぼし」の再取得については 1 時間に 1 回程度、最新データの同期処理を自動的に行います。

即座にデバイスの最新情報を取得したい場合にはこの API を実行する事で最新情報の取得が可能となります。但し、この API を高頻度に行うことはデバイスに大きく負荷を与える恐れがありますのでご注意ください。また、連続使用を行う場合は、前回実行時のレスポンスが返却され次第の実行を行ってください。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアル No です。

レスポンスサンプル

```
{
  "recogLogRegistered": 15,
  "recogPictureRegistered": 15,
  "operateLogRegistered": 23
}
```

レスポンス仕様

項目	型	概要
recogLogRegistered	数値	同期処理の結果、クラウドサーバーに保存された認識ログの件数(差分件数)です。一度あたり、最大で 100 件まで同期されます。
recogPictureRegistered	数値	同期処理の結果、クラウドサーバーに保存された認識写真画像の件数(差分件数)です。一度あたり、最大で 100 件まで同期されます。
operateLogRegistered	数値	同期処理の結果、クラウドサーバーに保存された操作ログの件数(差分件数)です。一度あたり、最大で 500 件まで同期されます。

APB 状態の修復

※本機能は facetCloud APB(アンチパスバック)ご加入者様のみ対象です

API URL	https://[HOST]/api1/device/apbRepaire
排他制御	有り (serialNo 単位に一リクエストのみを処理可能)

一時的なデバイスやネットワーク不調などよりリアルタイム APB 情報を取り逃したデバイスに対して、状況修復のための処理を実行します。

クラウドシステムは、APB に関するデータ通信に失敗した対象を保持しており、そのデータに基づいて修復を試みます。

修復の際には認識時刻が非常に重要な判定材料となるため、デバイスの時刻同期 (NTP) は必ずオンにして運用してください。

修復されるエラーケースは下記の 2 パターンです。

- ・認識したデバイス自体がネットワーク不調によりクラウドサーバーへデータを送信出来なかったケース
- ・認識デバイスと同一 APB グループに属しており、リアルタイムデータの受信が正常に行えなかったデバイスが存在したケース

修復は「人物ごと・デバイスごと」に実行されます。

過去の試行回数がより少ないデータから優先的に処理が行われ、一回あたり最大で 100 件までの修正が試行されます。

また、修正の試行は最大で「人物ごと・デバイスごと」に最大で 10 回です。これを超える問題については自動的な修復の対象外となります。

10 回を超え、自動修復の対象外となったデータについては、「クラウドサーバーへ人物の通行可能時間帯を登録(/person/registPersonAccessTimeForCloud)」の機能及び「クラウドサーバーに登録された人物データをデバイスに登録 (/person/toDevice)」を使用する事で修復を行う必要があります。

尚、この処理はクラウドサーバー側にて一定時間ごとに自動起動しており、通常はこの機能を利用する必要はありません。

早急にデータ修復を実行したい場合に限りご利用ください。

リクエストパラメータ

リクエストパラメータは不要です。実行すると APB 対応デバイスとして指定された全てのデバイスを対象として処理が実行されます。

レスポンスサンプル

```
{
  "XXXXXXXXXX": {
    "success": [
      "Yamada", "Suzuki"
    ],
    "error": []
  },
  "YYYYYYYYYYY": {
    "success": [],
    "error": []
  }
}
```

レスポンス仕様

項目	型	概要
連想配列キー	文字列	デバイスのシリアル No を示します。
success	配列（文字列）	キーが示すデバイスにおいて、APB 修復機能によってデータが修復され、正しいデータがデバイスへ登録された人物コードの配列です。
error	配列（文字列）	キーが示すデバイスにおいて、APB 修復機能によってデータの修復を試行したものの、通信エラーなどにより失敗した人物コードの配列です。

認証時のプッシュ通知先 URL の取得

API URL	https://[HOST]/api1/device/getPushUrl
排他制御	無し

facetCloud の「PUSH の通知機能」に設定されている通知先 URL を取得します。デバイスごとに取得を行う処理となります。

この API は serialNo を指定しますが、デバイスと通信を行うものではありません。

※PUSH の通知機能については、本書の「認識の通知（プッシュ通知）」を参照ください。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアル No です。

レスポンスサンプル（正常）

```
{
  "push_url": "https://test.com/testApi"
}
```

レスポンス仕様

項目	型	概要
push_url	文字列	指定されたデバイスに登録されている push URL です。

認証時のプッシュ通知先 URL の登録

API URL	https://[HOST]/api1/device/registPushUrl
排他制御	無し

facetCloud の「PUSH の通知機能」に使用される通知先 URL を登録します。取得と同様、デバイスごとに登録を行う処理となります。

この API は serialNo を指定しますが、デバイスと通信を行うものではありません。

※PUSH の通知機能については、本書の「認識の通知（プッシュ通知）」を参照ください。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアル No です。
pushUrl		文字列	登録する pushUrl を指定します。 ※既に登録されている場合は上書き登録となります。

※既に登録されている内容を削除したい場合は、パラメータ「push_url」を空で送信してください。

レスポンスサンプル（正常）

```
{
  "result": true
}
```

レスポンス仕様

項目	型	概要
result	真偽値	true：指定のデバイスに対して push_url が登録されたことを示します。

認識ログの取得

API URL	https://[HOST]/api1/log/getRecogLog
排他制御	無し

デバイスが顔認識したログを検索します。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN o です。
latest		真偽値	最新の一件のみを取得する場合には 1 を指定します。このパラメータが指定された場合、recogTimeFrom / recogTimeTo / personCode / pageNo で指定した内容は無視されます。
recogTimeFrom		日付時刻	認識時刻の FROM を指定します。 フォーマットは YYYY/MM/DD HH:MI:SS です。
recogTimeTo		日付時刻	認識時刻の TO を指定します。 フォーマットは YYYY/MM/DD HH:MI:SS です。
personCode		文字列	人物コードを指定します。
pageNo		数値	取得するページ番号を指定します。 一度に取得される行数は最大で 100 件となります。 仮に 101 件目から 200 件目までを取得したい場合には 2 を指定します。 省略した場合のデフォルト値は 1 となります。
pictureExpires		数値	画像取得 Cookie の期限を秒数で指定します。 10 から 300 の整数で指定します。 詳細は「クラウドサーバーに保存された画像の参照について」の項もご参照ください。
pictureAllowIp		文字列	画像ファイルに対し、アクセス元 IP アドレスを制限する事が出来ます。指定しない場合、全

			ての IP からの閲覧が許可されます。 このパラメータを指定する事により、画像ファイルに対して更なるセキュリティの向上が期待出来ます。
--	--	--	---

レスポンスサンプル

```
{
  "rows":150,
  "pages":2,
  "pictureCookie":{
    "CloudFront-Policy":"xxxxxxx",
    "CloudFront-Signature":"xxxxxxx",
    "CloudFront-Key-Pair-Id":"xxxxxxx",
  }
  "list":[
    {
      "id":"125",
      "time":"2020/08/27 12:03:05",
      "personCode":"F00001",
      "personName":"山田 太郎",
      "pictureUrl":"https://xxxx.xx/xxx/xxxx.jpg",
      "mask":1,
      "temp":36.2,
      "pass":true,
      "searchScore":98
    },
    {
      "id":"124",
      "time":"2020/08/27 12:01:52",
      "personCode":"F00002",
      "personName":"山田 次郎",
      "pictureUrl":"https://xxxx.xx/xxx/xxxx.jpg",
      "mask":2,
      "temp":36,
      "pass":false,
      "searchScore":97.4
    },
    <以下略>
  ]
}
```

レスポンス仕様

項目	型	概要
rows	数値	検索条件に該当したデータの件数です。latest で 1 が指定された場合はこの項目は返却されません。
pages	数値	検索条件に該当したページ数です。latest で 1 が

		指定された場合はこの項目は返却されません。 一ページあたり、最大で 100 件のデータが返却されます。
pictureCookie	配列	画像を参照するための Cookie の名称と値です。 詳細は「クラウドサーバーに保存された画像の参照について」の項をご参照ください。
list	配列	検索結果のデータです。並び順は id の降順です。
id	数値	ログを一意に特定する id です。この id はログの削除や、新たな人物の登録の際に使う事が出来ます。 デバイス側での解析処理が全て終わっていない状態（速報データ）の場合、null が返却される場合がありますが、顔が認識された後の、数秒後には格納されます。
time	日付時刻	認識された日時です。 YYYY/MM/DD HH:MI:SS の形式です。 デバイスの内部時計の時刻に基づいているため、実時間との差異が起こる場合があります。
personCode	文字列	認識された人物のコードです。未登録人物である場合には null となります。
personName	文字列	認識された人物の氏名です。未登録人物である場合には null となります。 また、個人名マスクをお申込みの場合には固定の文字列「***」となります。
pictureUrl	文字列	https から始まる画像参照のための URL です。 html の img タグなどにエスケープ処理不要で直接利用出来ます。この URL を参照するためには、pictureCookie の値が Cookie として設定されている必要があります。
mask	数値	マスク着用状態を示します。 0: 不明, 1: 有り, 2: 無し
temp	数値	体温を示します。 判定不能だった場合は null となります。
pass	真偽値	デバイスが通行許可と判定していた場合には true、通行が許可されなかった場合には false とな

		ります。
searchScore	数値	識別スコアです。この数値が高いほど信頼性の高い認識結果である事を示します。小数点以下第一位までの値で格納されます。
passFlagName	文字列	勤怠区分が登録されており、人物に付与されている場合の付与された勤怠区分です。登録等していない場合は NULL となります。
accessType	文字列	認証を行った際の認証方式です。顔やカードの二要素認証等の判別をすることが可能となります。 "Face": 顔認証, "Card": カード/QR 認証, "FaceAndCard": 顔・カード(2要素認証) "FirstCardAndFace": カード先固定・顔(2要素認証)
cardNo	文字列	認識された人物が認証をする際に使用した IC カード番号です。カード認証を行っていない場合は null となります。
cardType	文字列	認証されたカードのタイプです。IC カードか QR コードかを判別することが可能です。カード認証しないときは null になります。 1: IC, 6:QR
cardBindPersonCode	文字列	カード認証した際、カードで認識された人物のコードです。カード認証しないときは null になります。
cardBindPersonName	文字列	カード認証した際、カードで認識された人物の氏名です。カード認証しないときは null になります。

認識ログの削除

API URL	https://[HOST]/api1/log/deleteRecogLog
排他制御	有り（serialNo 単位に一リクエストのみを処理可能）

デバイスが認識したログを削除します。これは「論理削除」ではなく、いわゆる「物理削除」に該当する処理のため、一度削除すると元には戻せません。

デバイス本体とクラウドサーバーの両方からデータが削除されます。

これは情報セキュリティポリシーへの準拠や、個人情報上のトラブルへの対応を想定した機能となっております。そのため、もし誤ってログを削除してしまった場合の復元のお問い合わせには対応いたしかねますので、よくご理解の上でご利用下さい。

一度の API 実行では、最大で 100 件が削除されます。つまり仮に条件に合致するレコードが 1000 件の場合、この API を 10 回実行する事で目的のレコードを全て削除出来る事になります。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアル No です。
ids	△	数値（カンマ区切り）	ログを特定するための id です。 id は「認識ログの取得(/log/getRecogLog)」から得る事が出来ます。 カンマ区切りで最大で 100 件まで指定可能です。
recogTimeFrom	△	日付時刻	認識時刻の FROM を指定します。 フォーマットは YYYY/MM/DD HH:MI:SS です。
recogTimeTo	△	日付時刻	認識時刻の TO を指定します。 フォーマットは YYYY/MM/DD HH:MI:SS です。
personCode	△	文字列	人物コードを指定します。

※ △のパラメータのうち、どれか一つは必須です。複数の条件パラメータを指定した場合、AND で対象が検索されます。

レスポンスサンプル（正常）

```
{
  "result":true,
  "total":98,
  "deleted":98
}
```

レスポンスサンプル(一部は成功したが、通信エラーなどにより途中で失敗した場合)

```
{
  "result":false,
  "total":1500,
  "deleted":97,
  "message":"ログ ID[1156]について、デバイスとの通信異常により削除に失敗し、処理を中断しました。"
}
```

レスポンス仕様

項目	型	概要
result	真偽値	true:処理が正常に終了しています。 false:何かしらの問題により全ての処理が正常に終了しなかった場合に該当します。
total	数値	指定された検索条件に合致した認識ログのレコード数です。
deleted	数値	削除された認識ログの件数です。
message	文字列	エラーの内容です。result が false の場合にのみ、格納されます。

操作ログの取得

API URL	https://[HOST]/api1/log/getOperateLog
排他制御	無し

デバイスに対する操作ログを検索します。

操作ログは 30 分程度に一度、定時処理により自動同期されます。

そのため、最新の操作ログ情報がすぐに必要となる場合には「デバイス同期要求 (/device/sync)」を事前に実行して頂く必要があります。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアル No です。
operateTimeFrom		日付時刻	操作時刻の FROM を指定します。 フォーマットは YYYY/MM/DD HH:MI:SS です。
operateTimeTo		日付時刻	操作時刻の TO を指定します。 フォーマットは YYYY/MM/DD HH:MI:SS です。
operateUser		文字列	操作したユーザー名を指定します。例えば 「admin」などです。 クラウドサーバーやデバイスの管理機能のロ グの場合は、ユーザー名は固有名「System」と なります。
mainType		文字列	メインタイプを指定します。例えば「操作」な どです。日本語文字列のままで指定して下さ い。
subType		文字列	サブタイプを指定します。例えば「ユーザーロ グイン」などです。日本語文字列のままで指定 して下さい。
pageNo		数値	取得するページ番号を指定します。 一度に取得される行数は最大で 100 件となり ます。仮に 101 件目から 200 件目までを取得

			したい場合には2を指定します。 省略した場合のデフォルト値は1となります。
--	--	--	--

レスポンスサンプル

```
{
  "rows": 1054,
  "pages": 11,
  "list": [
    {
      "operateTime": "2020/10/09 18:52:04",
      "operateUser": "admin",
      "mainType": "操作",
      "subType": "ユーザーログイン",
      "detail": "ClientType:Web,RemoteIP:111.111.111.111"
    },
    {
      "operateTime": "2020/10/09 18:47:24",
      "operateUser": "System",
      "mainType": "操作",
      "subType": "起動",
      "detail": ""
    }
    ...
  ]
}
```

レスポンス仕様

項目	型	概要
rows	数値	検索条件に該当したデータの件数です。
pages	数値	検索条件に該当したページ数です。一ページあたり、最大で100件のデータが返却されます。
list	配列	検索結果のデータです。 並び順は操作日時の降順です。
operateTime	数値	操作日時です。 YYYY/MM/DD HH:MI:SS の形式です。
operateUser	日付時刻	操作したユーザーです。
mainType	文字列	操作内容のメインタイプ（種別）です。
subType	文字列	操作内容のサブタイプ（詳細種別）です。
detail	文字列	記録された操作内容の詳細です。

認識の通知（プッシュ通知）

push URL	管理 WebUI もしくは、API: registPushUrl にて設定ください。 ※可能な限り https を推奨致します。
排他制御	無し
発信元 IP アドレス	設定完了後にお知らせ致しますので、ファイヤーウォール等の設定が行われている場合には、インバウンドの許可を行ってください。
Method	POST

この機能は厳密には API では無く、ご指定サーバーへの通知機能となります。デバイスが顔を認識した際に、数秒以内にご指定の URL に通知を行います。

この機能は一時的な通信障害(オフライン)が発生すると、通知がされないなどの理由により、確実性を保証するものではありません。

つまりこの機能は、金銭のやり取りを伴うようなクリティカルな要件には適合しません。正規データとして認識データを蓄積する目的の場合には crontab などにより定期的に「認識ログの取得(/log/getRecogLog)」を実行する事でデータの取得/蓄積を行って下さい。

リクエストパラメータ ※API サーバー(http クライアント)⇒ご指定の URL(http サーバー)

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアル No です。
time	○	日付時刻	認識された日時です。 YYYY/MM/DD HH:MI:SS の形式です。 デバイスの内部時計の時刻に基づいているため、実時間との差異が起こる場合があります。
personCode		文字列	認識された人物のコードです。未登録人物である場合には null となります。
personName		文字列	認識された人物の氏名です。未登録人物である場合には null となります。 また、個人名マスクをお申込みの場合には固定の文字列「***」となります。

pictureUrl		文字列	https から始まる画像参照のための URL です。html の img タグなどにエスケープ処理不要で直接利用出来ます。画像 URL はワンタイムなものであり、外部からは推測不能なランダムな文字列で構成されます。 また、URL はこの API 実行後、2 分間程で再利用不能な形で無効化されます。
mask		数値	マスク着用状態を示します。 0: 不明, 1: 有り, 2: 無し
temp		数値	体温を示します。 判定不能だった場合は null となります。
pass		数値	自動ドアやゲートが接続されている場合は、開錠したか否かを示します。 1: 開錠(判定 OK) 1 以外: 開錠失敗(判定 NG)
searchScore		数値	最も類似度が高かった人物との類似度スコアとなります。
accessType		文字列	認証を行った際の認証方式です。顔やカードの二要素認証等の判別をすることが可能となります。 "Face": 顔認証, "Card": カード/QR 認証, "FaceAndCard": 顔・カード(2 要素認証) "FirstCardAndFace": カード先固定・顔(2 要素認証)
cardNo		文字列	認識された人物が認証をする際に使用した IC カード番号です。カード認証を行っていない場合は null となります。
cardType		数値	認証されたカードのタイプです。IC カードか QR コードかを判別することが可能です。カード認証しないときは null になります。 1: IC, 6: QR
cardBindPersonCode		文字列	カード認証した際、カードで認識された人物のコードです。カード認証しないときは null になります。

人物の取得

API URL	https://[HOST]/api1/person/getPerson
排他制御	無し

対象の人物データをデバイスとクラウドサーバーの両方から取得します。この機能は都度デバイスにアクセスを行います。この機能を使用する事で、デバイスに直接、人物を登録した場合などに実際の登録状況を知る事が出来ます。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN oです。
personCode	○	文字列	人物コードを指定します。
includePicture		真偽値	写真画像を取得するかどうかを指定します。 0: 写真画像を取得せずにデータのみを取得します。 1: 写真画像を取得します。 このパラメータを指定しない場合のデフォルトは0(画像を取得しない)です。 画像を利用する場合にのみ1を指定して下さい。 デバイスからの写真画像の取得処理には時間が掛かる場合がある上、レスポンスのデータ容量も大きくなるためです。

レスポンスサンプル

```
{
  "device": {
    "personCode": "test",
    "personName": "テスト",
    "sex": "male",
    "birthday": "1980/12/05",
    "picture": "<base64 エンコード済み jpeg>",
    "accessTimes": [
      {
        "accessFlag": 1,
        "accessTimeFrom": "2000¥/05¥/13 13:32:14",
        "accessTimeTo": "2001¥/05¥/14 13:32:14"
      },
      {
        "accessFlag": 0,
        "accessTimeFrom": "2021¥/05¥/13 16:57:53",
        "accessTimeTo": "2023¥/05¥/14 16:57:53"
      }
    ]
  },
  "cloud": {
    "personCode": "test",
    "personName": "テスト",
    "sex": "male",
    "memo": "XXXXXX",
    "birthday": "1980/12/05",
    "picture": "<base64 エンコード済み jpeg>",
    "accessTimes": [
      {
        "accessFlag": 1,
        "accessTimeFrom": "2021¥/01¥/01 00:00:00",
        "accessTimeTo": "2021¥/01¥/01 00:00:01"
      }
    ]
  }
}
```

レスポンス仕様

項目	型	概要
device	配列	デバイスから取得したデータを示します。クラウドサーバーにのみ存在しているデータの場合は null となります。
cloud	配列	クラウドサーバーから取得したデータを示します。デバイスにのみ存在しているデータの場合は null となります。

personCode	文字列	人物のコードです。
personName	文字列	人物の氏名です。
sex	文字列 (列挙)	male(男性) female (女性)をそれぞれ示します。
birthday	日付	YYYY/MM/DD の形式です。
memo	文字列	管理用のメモです。cloud 配列側にのみ、この値は格納されます。
picture	文字列	base64 エンコードされた人物写真のデータ (jpg 形式) です。includePicture パラメータで 1 を指定した場合のみ格納されます。1 を指定しなかった場合は null となります。
accessTimes	配列	通行可能時間帯のリストが格納されます。
accessFlag	真偽値	0 の場合、通行不能を示します。 1 の場合、通行許可を示します。
accessTimeFrom	日時	通行可否の期間の FROM です。 フォーマットは YYYY/MM/DD HH:MI:SS です。
accessTimeTo	日時	通行可否の期間の TO です。 フォーマットは YYYY/MM/DD HH:MI:SS です。

人物画像のチェック

API URL	https://[HOST]/api1/person/checkPersonPicture
排他制御	無し
Method	POST

人物の顔写真について、デバイスが認識可能な画像ファイルかどうかをチェックします。この機能ではクラウドサーバーやデバイスには画像ファイルが保存される事はありません。確認結果を返却するのみの動作となります。また、デバイスにはアクセスしないため、デバイスの電源を切った状態であっても、利用する事が出来ます。

この機能は主に、Web ページ上でアップロードされた顔写真について、利用可能な画像であるかどうかをリアルタイムに判定し、認識不能である場合にユーザーに別の画像のアップロードを促すために利用する事を想定しています。

リクエストパラメータ

項目	必須	型	概要
picture	○	文字列	人物の顔写真画像ファイルのバイナリを Base64 エンコードした文字列を指定します。 この画像をもとにデバイスは人物認証を行うため、可能な限り「正面からの」「鮮明な」「マスクや装飾品を付けていない」写真を選択してください。 画像ファイルは jpg 形式である必要があります。推奨される画像サイズは「幅：480px」「高さ：720px」です。 Base64 エンコード後のファイルサイズが 5MB 以内である必要があります。
algorithmVersion		文字列	顔認識アルゴリズムのバージョン情報を指定します。 現時点はサポートされていませんが、将来的に利用可能となる予定です。 複数台のデバイスをご契約頂き、尚且つそれぞれのデバイスのバージョンが大きく異なる場合に、このパラメータを指定して頂く事を想っています。

レスポンスサンプル(顔認識に利用可能な画像の場合)

```
{
  "result":true
}
```

レスポンスサンプル(顔認識に利用出来ない画像の場合)

```
{
  "result":false,
  "message": "アルゴリズム不明のエラーです。顔として認識出来ない写真の可能性があります。
別の顔写真を指定して下さい。"
}
```

レスポンス仕様

項目	型	概要
result	真偽値	true: 顔認識に利用出来る画像です。 false: 顔認識に利用出来ない画像です。クラウドサーバーに保存する事は出来ませんが、「クラウドサーバーに登録された人物データをデバイスに登録 (/person/toDevice)」機能を使いデバイスに人物を登録する際に、エラーが発生します。
message	文字列	result が false の場合に、エラーの内容が格納されます。

クラウドサーバーに登録された人物の検索

API URL	https://[HOST]/api1/person/getPersonFromCloud
排他制御	無し

クラウドサーバーに登録されている人物のリストを取得します。この機能は、デバイスにはアクセスしないため、デバイスが稼働していなくても利用可能な上、高速に動作します。

人物データはクラウドサーバーではデバイス縦断で（ご契約の API トークンの単位で）データが管理されます。

もし、API 経由では無く、デバイスの管理ツールから直接人物データを登録している場合には、この機能でその人物が返却されない場合があります。デバイスに直接登録したデータをクラウドサーバーで利用出来るようにするためには、「デバイスに登録された人物データをクラウドサーバーに登録（/person/toCloud）」を実行して下さい。

リクエストパラメータ

項目	必須	型	概要
personCode		文字列	検索条件として人物コードを指定します。 アスタリスク記号の利用により、前方一致検索と後方一致検索がサポートされます。 例) PERSON_0001*
personName		文字列	検索条件として人物名を指定します。 アスタリスク記号の利用により、前方一致検索と後方一致検索がサポートされます。 例) 後藤*
personTypeCode		数値	検索条件として区分コードを指定します。 登録されている区分を確認する場合は「getPersonType」を使用してください。 ※入退管理モード(オプション)を利用していない場合は値を指定しても無視されます。
registSerialNo		文字列	検索条件としてシリアル No を指定します。このデバイスに登録されている（とクラウドサーバーが認識している）人物のみを抽出します。
pageNo		数値	取得するページ番号を指定します。 一度に取得される行数は最大で 100 件となります。仮に 101 件目から 200 件目までを取得した

			い場合には 2 を指定します。 省略した場合のデフォルト値は 1 となります。
pictureExpires		数値	画像取得 Cookie の期限を秒数で指定します。10 から 300 の整数で指定します。 詳細は「クラウドサーバーに保存された画像の参照について」の項もご参照ください。
pictureAllowIp		文字列	画像ファイルに対し、アクセス元 IP アドレスを制限する事が出来ます。指定しない場合、全ての IP からの閲覧が許可されます。 このパラメータを指定する事により、画像ファイルに対して更なるセキュリティの向上が期待出来ます。

レスポンスサンプル

```
{
  "rows":150,
  "pages":2,
  "pictureCookie":{
    "CloudFront-Policy":"xxxxxxx",
    "CloudFront-Signature":"xxxxxxx",
    "CloudFront-Key-Pair-Id":"xxxxxxx",
  },
  "list":[
    {
      "personCode":"F0002",
      "personName":"山田 太郎",
      "pictureUrl":"https://xxxx.xx/xxx/xxxx.jpg",
      "sex":"male",
      "memo":"管理メモです",
      "personTypeName":"テスト区分1",
      "":"1982/12/06",
      "serialNos":["XXXXX01","XXXXX02"]
    },
    {
      "personCode":"F0001",
      "personName":"山田 次郎",
      "pictureUrl":"https://xxxx.xx/xxx/xxxx.jpg",
      "sex":"male",
      "memo":"管理メモです",
      "birthday":"1980/12/06",
      "personTypeName":"テスト区分2",
      "serialNos":["XXXXX01"]
    }
  ]
  <以下略>
}
```

レスポンス仕様

項目	型	概要
rows	数値	検索条件に該当したデータの件数です。
pages	数値	検索条件に該当したページ数です。一ページあたり、最大で 100 件のデータが返却されます。
pictureCookie	配列	画像を参照するための Cookie の名称と値です。画像を参照するためには、画像 URL にアクセスする際に、ここに格納される Cookie が設定されている必要があります。Cookie の値はワンタイムなものであり、再利用は有効期限の時間内に限定されます。詳細は「クラウドサーバーに保存された画像の参照について」の項をご

		参照ください。
list	配列	検索結果のデータです。 並び順は、「登録された順」の降順です。
personCode	文字列	人物のコードです。
personName	文字列	人物の氏名です。
pictureUrl	文字列	https から始まる画像参照のための URL です。html の img タグなどにエスケープ処理不要で直接利用出来ます。 この URL を参照するためには、pictureCookie の 値が Cookie として設定されている必要があります。
sex	文字列 (列挙)	male(男性) female (女性)をそれぞれ示します。
birthday	日付	YYYY/MM/DD の形式です。
personTypeName	文字列	人物に付与されている区分名です。 入退管理モード(オプション)を利用していない場合は 「NULL」となります。
serialNos	配列 (文字列)	この人物が登録されているデバイスのシリアル No の 配列です。 この値の整合性は、クラウドサーバーを経由して操作 された範囲に限ります。 つまり、デバイスに直接人物を登録した場合や、デバイ スを工場出荷時に戻した場合など、クラウドサーバー が関与していない操作については、それらのシリアル No の正確な登録状況はこの配列には反映されません。 もし、この値に不整合が起きており、正確なものに最新 化したい場合には、「人物の取得(/person/getPerson)」 機能呼び出す事で、実際の状況と一致するようにク ラウドサーバー側のデータが更新されます。

クラウドサーバーへ人物を登録

API URL	https://[HOST]/api1/person/registPersonForCloud
排他制御	無し
Method	POST

人物をクラウドサーバーに登録します。人物コード（personCode）がまだ存在しなければ、新規に作成されます。既に存在する場合には、データが更新されます。

この機能は他の多くの API とは異なり、デバイスのシリアル No を指定する必要がありません。デバイス縦断で（ご契約の API トークンの単位で）データが管理されます。

また、この機能を使うだけでは、まだデバイスで顔認証を利用する事は出来ません。

デバイスに人物データを反映するためには「クラウドサーバーに登録された人物データをデバイスに登録（/person/toDevice）」を実行する必要があります。

リクエストパラメータ

項目	必須	型	概要
personCode	○	文字列	人物コードを指定します。
personName		文字列	人物名を指定します。データ上、姓名は分割されていませんので、フルネームで一つの項目です。 また、ご契約時に「個人情報を保存しない」を選択された場合、このパラメータにどのような値を設定したとしても、クラウドサーバーにはマスクされた値「***」で登録されます。
sex		文字列（列挙）	male(男性) female（女性）をそれぞれ示します。 顔認識アルゴリズムに影響を出す項目ではなく、管理用の項目です。 必須では無くクラウドには空（null）として保存できますが、デバイスの仕様上、空（null）の場合には male として保存されます。
birthday		日付	YYYY/MM/DD の形式です。 顔認識アルゴリズムに影響を出す項目ではなく、管理用の項目です。 必須では無くクラウドには空（null）として保存できますが、デバイスの仕様上、空（null）の場合には 1970/01/01 として保存されます。
memo		文字列	管理用のメモです。500 文字までで、任意の文

			字列を指定して下さい。
personTypeCode		数値	区分コードを指定します。区分コードを確認する場合は「getPersonType」をご利用ください。 ※入退管理モード(オプション)を利用していない場合、値を指定しても無視されます。
recogLogSerialNo	△	文字列	人物の顔写真を認識ログから自動取得する場合に指定をしてください。
recogLogId	△	数値	人物の顔写真を認識ログから自動取得する場合に指定をしてください。 このログ ID は「認識ログの取得 (/log/getRecogLog)」のレスポンスから得る事が出来ます。
picture	△	文字列	人物の顔写真画像ファイルのバイナリを Base64 エンコードした文字列を指定します。 この画像をもとにデバイスは人物認証を行うため、可能な限り「正面からの」「鮮明な」「マスクや装飾品を付けていない」写真を選択してください。 画像ファイルは jpg 形式である必要があります。推奨される画像サイズは「幅：480px」「高さ：720px」です。 Base64 エンコード後のファイルサイズが 5MB 以内である必要があります。

※ 「recogLogSerialNo 及び recogLogId」「picture」のどちらかは必須です。両方指定した場合には picture が優先されます。

レスポンスサンプル

```
{
  "result": true
}
```

レスポンス仕様

項目	型	概要
result	真偽値	true: クラウドサーバーに人物情報が登録された事を示します。

クラウドサーバーに登録された人物データをデバイスに登録

API URL	https://[HOST]/api1/person/toDevice
排他制御	有り（serialNo 単位に一リクエストのみを処理可能）

クラウドサーバーに登録された人物データをデバイスにダウンロードし、人物として登録します。これにより、対象のデバイスで実際に顔認証が利用出来るようになります。

デバイスに人物データを登録する際、画像ファイルに対して解析処理が行われます。顔認識不能な画像だった場合にこの処理はエラーを起こし、登録に失敗します。適切な画像ファイルであるかどうかを事前にチェックするには「人物画像のチェック (/person/checkPersonPicture)」を利用して下さい。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN oです。
personCode	○	文字列	人物コードを指定します。 ※該当する人物の personName がない場合はエラーが発生し、登録できません。
override		真偽値	同一の人物コードが既にデバイスに存在していた場合の挙動を指定します。 0:クラウド側のデータで上書きを行わずに処理を中断します。 1:クラウド側のデータで上書きを行います。 指定しない場合のデフォルトは 1（上書く）です。
noImage		真偽値	デバイスにデータを送信する際に、画像データを含むか否かを指定します。 0:画像データを含めます。 1:画像データを含めずに人名や通行可能時間帯データのみを登録します。画像データを変更しない場合にはこちらの方が低い負荷で動作します。 指定しない場合のデフォルトは 0(画像データを含める)です。 まだデバイスに登録されていない人物のデータをデバイスに新規に登録する場合、このパラメータは指定無し、もしくは 0 を指定して下さい。

レスポンスサンプル（登録成功時）

```
{
  "result": true,
  "regist": true,
  "message": "人物がデバイスに登録されました。"
}
```

レスポンスサンプル（上書きしない事を指定し既にデータがデバイスに存在している場合）

```
{
  "result": true,
  "regist": false,
  "message": "指定された personCode に該当する人物データがデバイスに登録されており、尚  
且つ override で 0 が指定されているため、登録を取りやめました。"
}
```

レスポンスサンプル（デバイスが顔写真を認識出来なかった場合など）

```
{
  "result": false,
  "regist": false,
  "message": "アルゴリズム不明のエラーです。顔として認識出来ない写真の可能性がありま  
す。別の顔写真を指定して下さい。"
}
```

レスポンスサンプル（解析の結果、同一人物と判定された場合）

```
{
  "result": false,
  "regist": false,
  "message": "既に保存されている人物の顔写真です。"
}
```

※personCode が異なるデータであっても、人物の画像を解析した結果、既にデバイスに登録済みの人物と同一人物であると判定された場合、このエラーが起こる場合があります。

レスポンス仕様

項目	型	概要
result	真偽値	処理が意図通りに行われたかが格納されます。 true: 処理が正常に終了した事を示します。 false: 写真の認識失敗などにより、処理が正常に終了していない事を示します。

		override パラメータに 0(上書きしない)を指定し、尚且つ既にデバイスに人物データが登録されていた場合、意図通りの動作とみなし、result には true が格納されます。
regist	真偽値	デバイスに実際に人物データが登録されたかどうか格納されます。 true: デバイスに人物データが登録された事を示します。 false: デバイスに人物データが登録されなかった事を示します。 override パラメータに 0(上書きしない)を指定し、尚且つ既にデバイスに人物データが登録されていた場合、regist には false が格納されます。
message	文字列	処理結果のメッセージが格納されます。

クラウドサーバーに登録された人物データの削除

API URL	https://[HOST]/api1/person/deletePersonFromCloud
排他制御	無し

クラウドサーバーから人物データを削除します。これは「論理削除」ではなく、いわゆる「物理削除」に該当する処理です。デバイス上の人物データは削除されず、引き続き利用可能です。

リクエストパラメータ

項目	必須	型	概要
personCode	○	文字列	人物コードを指定します。

レスポンスサンプル

```
{
  "result":true
}
```

レスポンス仕様

項目	型	概要
result	真偽値	true:処理が正常に終了した事を示します。

クラウドサーバーに登録された人物とデバイスの関連付けの削除

API URL	https://[HOST]/api1/person/deletePersonAssociation
排他制御	無し

クラウドサーバーから人物とデバイスの関連付けを削除します。これは「論理削除」ではなく、いわゆる「物理削除」に該当する処理です。クラウドサーバー上で行われる処理のため、デバイスとの通信は行われません。

デバイスとの通信切断により処理が意図せぬ形で中断してしまった場合や、デバイス上で直接人物データを削除するとクラウドサーバー側とのデータ不整合が発生する場合があります。

その場合、この機能を使用する事により不整合を解消する事が可能となります。

リクエストパラメータ

項目	必須	型	概要
personCode	○	文字列	人物コードを指定します。
serialNo		文字列	デバイスのシリアルNoです。指定しなかった場合、全てのデバイスとの関連付けを削除します。

レスポンスサンプル

```
{
  "deleted":1
}
```

レスポンス仕様

項目	型	概要
deleted	数値	削除された関連付けの件数です。 仮に3つのデバイスとの関連付けが削除された場合、3となります。 関連付けを一切持たない人物データに対してこの機能が実行された場合には0となります。

クラウドサーバーに登録された人物の通行可能時間帯の取得

API URL	https://[HOST]/api1/person/getPersonAccessTimeFromCloud
排他制御	無し

クラウドサーバーに登録されている人物の通行可能時間帯リストを取得します。この機能は、デバイスにはアクセスしないため、デバイスが稼働していなくても利用可能な上、高速に動作します。

リクエストパラメータ

項目	必須	型	概要
personCode	○	文字列	人物コードを指定します。

レスポンスサンプル

```
{
  "AAAAAAAAA1": [
    {
      "accessFlag": 1,
      "accessTimeFrom": "2021¥/01¥/01 00:01:01",
      "accessTimeTo": "2021¥/01¥/01 00:01:01"
    },
    {
      "accessFlag": 0,
      "accessTimeFrom": "2022¥/01¥/02 00:11:01",
      "accessTimeTo": "2099¥/12¥/31 23:59:59"
    }
  ],
  "AAAAAAAAA2": [
    {
      "accessFlag": 1,
      "accessTimeFrom": "2021¥/01¥/01 00:01:01",
      "accessTimeTo": "2021¥/01¥/01 00:01:01"
    }
  ]
}
```

レスポンス仕様

項目	型	概要
<最上位連想配 列キー>	文字列	シリアル No を示します。
accessFlag	真偽値	0 の場合、通行不能を示します。

		1 の場合、通行許可を示します。
accessTimeFrom	日時	通行可否の期間の FROM です。 フォーマットは YYYY/MM/DD HH:MI:SS です。
accessTimeTo	日時	通行可否の期間の TO です。 フォーマットは YYYY/MM/DD HH:MI:SS です。

クラウドサーバーに登録された人物へ通行可能時間帯の登録

API URL	https://[HOST]/api1/person/registPersonAccessTimeForCloud
排他制御	無し

人物の通行可能時間帯リストをクラウドサーバーに登録します。

また、この機能を使うだけでは、まだデバイスには通行可能情報は登録されません。

デバイスに通行可能情報を反映するためには「クラウドサーバーに登録された人物データをデバイスに登録 (/person/toDevice)」を実行する必要があります。

リクエストパラメータ

項目	必須	型	概要
personCode	○	文字列	人物コードを指定します。
serialNo	○	文字列	制御対象のデバイスのシリアル No を指定します。
accessFlag_[N]		真偽値	0 の場合、通行不能とします。 1 の場合、通行許可とします。
accessTimeFrom_[N]	※	日時	対象となる期間の FROM を指定します。 yyyy/mm/dd hh:mi:ss の形式で指定します。 2020/01/01 00:00:00 から 2099/12/31 23:59:59 までが許容されます。
accessTimeTo_[N]	※	日時	対象となる期間の TO を指定します。 yyyy/mm/dd hh:mi:ss の形式で指定します。 2020/01/01 00:00:00 から 2099/12/31 23:59:59 までが許容されます。

※同一[N]の accessFlag が指定されている場合に必須。

[N]の部分には 1 を起算とする連番を指定します。

例えば 3 件の通行許可情報を登録する場合には下記のようなリクエスト構成となります。

```
https://[HOST]/api1/person/registPersonAccessTimeForCloud
?personCode=xxxx&serialNo=XXXXXX
&accessFlag_1=1&accessTimeFrom_1=2021/05/03 10:00:00&accessTimeTo_1=2021/05/04 10:00:00
&accessFlag_2=1&accessTimeFrom_2=2021/05/05 10:00:00&accessTimeTo_2=2021/05/06 10:00:00
&accessFlag_3=1&accessTimeFrom_3=2021/05/07 10:00:00&accessTimeTo_3=2021/05/08 10:00:00
```

※実際のパラメータ値は適切な URL エンコードが行われている必要があります。

最大で 10 件までが許容され、11 件目以降は指定しても無視されます。

また、期間の重複指定は許容されません。

設定済みの通行可能時間帯情報を全て削除したい場合には、下記のように accessFlag をはじめとした通行時間帯関連のパラメータを指定せずにリクエストを行って下さい。

```
https://[HOST]/api1/person/registPersonAccessTimeForCloud
?personCode=xxxx&serialNo=XXXXXX
```

レスポンスサンプル

```
{
  "result": true
}
```

レスポンス仕様

項目	型	概要
result	真偽値	true: クラウドサーバーに人物の通行可能時間情報が登録された事を示します。

デバイスに登録された人物データの検索

API URL	https://[HOST]/api1/person/ getPersonFromDevice
排他制御	無し

デバイスの管理ツールから直接登録された人物データを検索します。この機能呼び出すと、その都度、デバイスにアクセスするため処理に時間が掛かる場合があります。

また、デバイスの仕様上、画像データを一括取得する事は負荷が掛かるため、この機能では顔写真の画像ファイルは返却されません。デバイスにのみ保存された顔写真が必要となる場合には、人物ごと（personCode ごと）に、「人物の取得(/person/getPerson)」機能を使い取得して下さい。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN oです。
personCode		文字列	検索条件となる人物コードを指定します。全後方一致で検索されます。※
personName		文字列	検索条件となる人物名を指定します。全後方一致で検索されます。※
pageNo		数値	取得するページ番号を指定します。 一度に取得される行数は最大で 100 件となります。 仮に 101 件目から 200 件目までを取得したい場合には 2 を指定します。 省略した場合のデフォルト値は 1 となります。

※ 検索方法について、「クラウドサーバーに登録された人物の検索

(/person/getPersonFromCloud)」機能とは微妙に異なる点にご注意下さい。アスタリスクはサポートされず、全て全後方一致となります。

レスポンスサンプル

```
{
  "rows": 2,
  "pages": 1,
  "list": [
    {
      "personCode": "X0002",
      "personName": "デバイス花子",
      "sex": "female",
      "birthday": "1970/01/01",
      "accessTimes": [
        {
          "accessFlag": 1,
          "accessTimeFrom": "2000¥/05¥/13 13:32:14",
          "accessTimeTo": "2001¥/05¥/14 13:32:14"
        },
        {
          "accessFlag": 0,
          "accessTimeFrom": "2021¥/05¥/13 16:57:53",
          "accessTimeTo": "2023¥/05¥/14 16:57:53"
        }
      ]
    },
    {
      "personCode": "X0001",
      "personName": "山田 太郎",
      "sex": "male",
      "birthday": "1982/01/01",
      "accessTimes": []
    }
  ]
}
```

レスポンス仕様

項目	型	概要
rows	数値	検索条件に該当したデータの件数です。
pages	数値	検索条件に該当したページ数です。一ページあたり、最大で 100 件のデータが返却されます。
list	配列	検索結果のデータです。 並び順は、「登録された順」の降順です。
personCode	文字列	人物のコードです。
personName	文字列	人物の氏名です。
sex	文字列 (列挙)	male(男性) female (女性)をそれぞれ示します。
birthday	日付	YYYY/MM/DD の形式です。

accessTimes	配列	通行可能時間帯のリストが格納されます。
accessFlag	真偽値	0 の場合、通行不能を示します。 1 の場合、通行許可を示します。
accessTimeFrom	日時	通行可否の期間の FROM です。 フォーマットは YYYY/MM/DD HH:MI:SS です。
accessTimeTo	日時	通行可否の期間の TO です。 フォーマットは YYYY/MM/DD HH:MI:SS です。

デバイスに登録された人物データをクラウドサーバーに登録

API URL	https://[HOST]/api1/person/toCloud
排他制御	無し

デバイスの管理ツールから直接人物データを登録した場合、クラウドサーバーはそのデータを自動では取得しません。デバイスにのみ、登録されたデータをクラウドサーバーに登録するためには、この機能を使い、対象の人物をクラウドサーバーにアップロードする必要があります。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN oです。
personCode	○	文字列	人物コードを指定します。
override		真偽値	同一の人物コードが既にクラウドサーバーに存在していた場合の挙動を指定します。 0:デバイス側のデータで上書きを行わずに処理を中断します。 1:デバイス側のデータで上書きを行います。 指定しない場合のデフォルトは1（上書く）です。

レスポンスサンプル（登録成功時）

```
{
  "result": true,
  "regist": true,
  "message": "人物がクラウドに登録されました。"
}
```

レスポンスサンプル（上書きしない事を指定し既にデータがクラウドに存在している場合）

```
{
  "result": true,
  "regist": false,
  "message": "指定された personCode に該当する人物データがクラウドに登録されており、尚  
且つ override で 0 が指定されているため、登録を取りやめました。"
}
```

レスポンスサンプル（デバイスに人物が登録されていない場合など）

```
{
  "result": false,
  "regist": false,
  "message": "指定された personCode に該当する人物データがデバイス側に登録されていま  
せん。"
}
```

レスポンス仕様

項目	型	概要
result	真偽値	処理が意図通りに行われたかどうか格納されます。 true: 処理が正常に終了した事を示します。 false: デバイスからの取得失敗などにより、処理が正常に終了していない事を示します。 override パラメータに 0(上書きしない)を指定し、尚且つ既にクラウドサーバーに人物データが登録されていた場合、意図通りの動作とみなし、result には true が格納されます。
regist	真偽値	クラウドサーバーに実際に人物データが登録されたかどうか格納されます。 true: クラウドサーバーに人物データが登録された事を示します。 false: クラウドサーバーに人物データが登録されなかった事を示します。 override パラメータに 0(上書きしない)を指定し、尚且つ既にクラウドサーバーに人物データが登録されていた場合、regist には false が格納されます。
message	文字列	処理結果のメッセージが格納されます。

デバイスに登録された人物データの削除

API URL	https://[HOST]/api1/person/deletePersonFromDevice
排他制御	有り（serialNo 単位に一リクエストのみを処理可能）

デバイスから人物データを削除します。これは「論理削除」ではなく、いわゆる「物理削除」に該当する処理です。クラウドサーバー上の人物データは削除されず、引き続き利用可能です。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN oです。
personCode	○	文字列	人物コードを指定します。

レスポンスサンプル

```
{
  "result":true
}
```

レスポンス仕様

項目	型	概要
result	真偽値	true:処理が正常に終了。

デバイスに登録された全人物データの削除

API URL	https://[HOST]/api1/person/clearPersonFromDevice
排他制御	有り（serialNo 単位に一リクエストのみを処理可能）

デバイスに登録された全ての人物データを削除します。クラウドサーバー側のデータは削除されません。

処理対象のデータ件数が多い場合、処理に時間が掛かる場合があります。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN oです。

レスポンスサンプル(正常に終了)

```
{
  "result":true
}
```

レスポンスサンプル(エラーが発生した場合)

```
{
  "result":false,
  "deleted":97,
  "message":"97 件まで削除されましたが、デバイスとの通信異常により削除に失敗し、処理を中断しました。"
}
```

レスポンス仕様

項目	型	概要
result	真偽値	true:処理が正常に終了（全件の削除が完了）。 false:何かしらの問題により処理が正常に終了しなかった。
deleted	数値	削除された認識ログの件数です。
message	文字列	エラーの内容です。result が false の場合にのみ、格納されます。

クラウドサーバーに登録された人物へカード情報の登録

API URL	https://[HOST]/api1/person/registPersonCardInfoForCloud
排他制御	無し

クラウドサーバーに登録されているユーザーに対して、カード情報の登録を行います。

最大で3つまでのカード登録が可能です。

また、この機能を使うだけでは、まだデバイスにはカード情報は登録されません。

デバイスにカード情報を反映するためには「クラウドサーバーに登録された人物データをデバイスに登録 (/person/toDevice)」を実行する必要があります。

リクエストパラメータ

項目	必須	型	概要
personCode	○	文字列	IC カード情報の登録対象人物の指定です。
card_no_[N]	○	文字列	IC カード番号です。 入力は半角英数字で英字は a~f (大文字小文字区別せず) での入力となります。 ※2
validityDateFrom_[N]	※1	日付	IC カード情報の有効期限の開始です。 yyyy/mm/dd の形式で入力 (To も同様)
validityDateTo_[N]	※1	日付	IC カード情報の有効期限の終了です。

※1 IC カード番号を入力した card_no_[N]の有効期限は必須となります。

※2 入力可能なカード番号の桁数について「16 進数の場合は 16 桁まで」、「10 進数の場合は 20 桁且つ 18446744073709551615 まで」となります。

例えば3件のカード情報を登録する場合には下記のようなリクエスト構成となります。

```
https://[HOST]/api1/person/ registPersonCardInfoForCloud
?ds-api-token=xxxxxx&personCode=xxxx
&card_no_1=yyy&validityDateFrom_1=2021/05/03&validityDateTo_1=2021/05/04
&card_no_2=zzz&validityDateFrom_2=2021/05/03&validityDateTo_2=2022/05/04
&card_no_3=ccc&validityDateFrom_3=2019/05/03&validityDateTo_3=2021/05/04
```

※実際のパラメータ値は適切な URL エンコードが行われている必要があります。

最大で3件までが許容され、4件目以降は指定しても無視されます。

設定済みのカード情報を全て削除したい場合には、下記のように card_no をはじめとした

カード情報のパラメータを指定せずにリクエストを行って下さい。

```
https://[HOST]/api1/person/registPersonCardInfoForCloud
?ds-api-token=xxxxxx&personCode=xxxx
```

レスポンスサンプル(正常時（人物登録時も同様）)

```
{
  "result": true
}
```

レスポンス仕様

項目	型	概要
result	真偽値	true:登録が正常に終了。 false:何かしらの問題により登録が正常に終了しなかった。
message	文字列	エラーの内容です。result が false の場合にのみ、格納されます。

クラウドサーバーに登録された人物のカード情報の取得

API URL	https://[HOST]/api1/person/getPersonCardInfoFromCloud
排他制御	無し

クラウドサーバーに登録されている人物のカード情報(最大3つまで)を取得します。この機能は、デバイスにはアクセスしないため、デバイスが稼働していなくても利用可能な上、高速に動作します。

リクエストパラメータ

項目	必須	型	概要
personCode	○	文字列	人物コードを指定します。

レスポンスサンプル

```
{
  "list": [
    {
      "card_no": "111",
      "validityDateFrom": "2021¥/01¥/01",
      "validityDateTo": "2034¥/01¥/01"
    },
    {
      "card_no": "222",
      "validityDateFrom": "2021¥/01¥/01",
      "validityDateTo": "2021¥/01¥/01"
    },
    {
      "card_no": "333",
      "validityDateFrom": "2015¥/01¥/01",
      "validityDateTo": "2021¥/01¥/01"
    }
  ]
}
```

レスポンス仕様

項目	型	概要
list	配列	登録されているカード情報の配列を表します。
validityDateFrom	日時	IC カード情報の有効期限の開始です。
accessTimeTo	日時	IC カード情報の有効期限の終了です。

人物の顔の撮影と撮影画像の取得

API URL	https://[HOST]/api1/person/capturePersonPicture
排他制御	有り

デバイスをキャプチャモードに移行させて、撮影した顔写真を取得します。撮影の許容時間(タイムアウトまでの時間)を `getCount` で指定します。

※当 API はデバイスの動作が伴うため、連続の使用をする場合は、前回のレスポンスが返却され次第、次の API を実行してください。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN oです。
getCount		数値(秒数)	撮影時間 1～30 秒で指定できます。何も指定しない場合は 30 秒で動作します。

レスポンスサンプル(正常時 (人物登録時も同様))

<jpg 画像のバイナリが返却されます>

レスポンスサンプル (画像取得ができなかった場合)

```
{
  "result": false,
  "message": "顔画像が取得出来ませんでした。"
}
```

デバイスに登録された人物画像との類似度チェック

API URL	https://[HOST]/api1/person/checkSimilarityInDevice
排他制御	無し
Method	POST

顔画像を指定して、デバイス内で登録されている人物画像の中から、最も類似度が高い人物の ID とそのスコアを返却します。

デバイス制限

本機能は 1200/TACX1250/TACX1270 の FWver85 以降でご利用いただけます。

※TTCX1100/5MA0721-E-JD05 では本機能はご利用いただけません。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアル No です。
picture	○	文字列	人物の顔写真画像ファイルのバイナリを Base64 エンコードした文字列を指定します。 画像ファイルは jpg 形式である必要があります。推奨される画像サイズは「幅：480px」「高さ：720px」です。 Base64 エンコード後のファイルサイズが 5MB 以内である必要があります。

レスポンスサンプル（正常時）

```
{
  "result": true
  "personCode":12
  "score":93.1
}
```

レスポンスサンプル（チェックできなかった場合）

```
{
  "result": false,
  "message": "チェック画像に顔が複数存在します。"
}
```

レスポンス仕様

項目	型	概要
result	真偽値	true: 類似度チェックができたことを示します。

		false: 類似度チェックができなかったことを示します。
personCode	文字列	最も類似度が高かった人物の ID となります。
score	数値	最も類似度が高かった人物との類似度スコアとなります。
message	文字列	result が false となった原因となります。

デバイスに登録された人物画像の取得

API URL	https://[HOST]/api1/person/getPersonPictureFromDevice
排他制御	無し
Method	POST

デバイスに登録されている人物の顔画像を取得します。personCode での指定で人物の特定を行います。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアル No です。
personCode	○	文字列	デバイスに登録されている人物の ID です。

レスポンスサンプル(正常取得時)

<jpg 画像のバイナリが返却されます>

レスポンスサンプル（指定した ID がデバイス内に存在しなかった場合）

```
{
  "result": false,
  "message": "該当の personCode が存在しません。"
}
```

レスポンス仕様

項目	型	概要
result	真偽値	false: デバイス内に指定 ID の人物が存在しなかった場合。
message	文字列	エラーの内容です。result が false の場合にのみ、格納されます。

クラウドサーバーに登録された区分の取得

API URL	https://[HOST]/api1/person/getPersonType
排他制御	無し

クラウドサーバーに登録されている区分を取得します。

この機能は他の多くの API とは異なり、デバイスのシリアル No を指定する必要がありません。ご契約の API トークンの単位でデータ取得がされます。

※この API は入退管理モードにてご契約されている場合のみ使用できます。

リクエストパラメータ

なし

レスポンスサンプル(正常取得時)

```
[
  {
    "person_type_code": "1",
    "person_type_name": "区分 1"
  },
  {
    "person_type_code": "2",
    "person_type_name": "区分 2"
  }
]
```

レスポンスサンプル（入退管理モードを利用していない環境の場合）

```
{
  "error": true,
  "errors": [
    {
      "name": "enter_exit_mode_flag",
      "message": "入退管理モード以外ではこの API は実行できません。"
    }
  ]
}
```

レスポンス仕様

項目	型	概要
person_type_code	数値	登録されている区分を識別する値となります。
person_type_name	文字列	登録されている区分名となります。

クラウドサーバーへ区分の登録

API URL	https://[HOST]/api1/person/registPersonType
排他制御	無し

クラウドサーバーに区分を登録します。既に存在する区分 CODE(personTypeCode)を指定した場合は区分名が更新されます。

この機能は他の多くの API とは異なり、デバイスのシリアル No を指定する必要がありません。ご契約の API トークンの単位でデータ管理がされます。

※区分の登録上限は「10」となります。それ以上を超えた登録はエラーとなります。

※この API は入退管理モードにてご契約されている場合のみ使用できます。

リクエストパラメータ

項目	必須	型	概要
personTypeCode	△	数値	登録する区分を識別するためのコードとなります。指定しない場合は自動で採番されます。既に存在する personTypeCode を指定し、override を「1」にすることで上書き登録を行います。
personTypeName	○	文字列	登録する区分名となります。
override	△	数値	上書き登録を行いたい場合は「1」を指定します。存在しない personTypeCode を指定している場合は override に関わらず新規登録となります。

レスポンスサンプル(正常登録)

```
{
  "result": true
}
```

レスポンスサンプル（登録上限数を超えて登録しようとした場合）

```
{
  "result": false,
  "message": "登録上限数「10」に達しています。"
}
```

レスポンス仕様

項目	型	概要
result	真偽値	true:区分が問題なく登録できたことを示します。 false:区分登録上限数に達している場合。
message	文字列	エラーの内容です。result が false の場合にのみ、格納されます。

クラウドサーバーに登録された区分の削除

API URL	https://[HOST]/api1/person/deletePersonType
排他制御	無し

クラウドサーバーに登録されている区分を削除します。指定した personTypeCode の区分を削除します。

この機能は他の多くの API とは異なり、デバイスのシリアル No を指定する必要がありません。ご契約の API トークンの単位でデータ管理がされます。

削除を行った場合、削除した区分にて既に登録されている人物については、区分は解除され、認証ログについても区分がない状態へ戻ります。(人物やログ自体は削除されません)

※この API は入退管理モードにてご契約されている場合のみ使用できます。

リクエストパラメータ

項目	必須	型	概要
personTypeCode	○	数値	削除する区分コードとなります。 区分コードを確認する場合は「getPersonType」を実行してください。

レスポンスサンプル

```
{
  "result": true
}
```

レスポンス仕様

項目	型	概要
result	真偽値	true:区分の削除及び削除した区分との関連も解除されたことを示します。

設定の取得

API URL	https://[HOST]/api1/config/getConfig
排他制御	無し

デバイスの設定情報を取得します。

この API を実行すると、その都度デバイスにアクセスし、最新の設定データを取得します。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN oです。

レスポンスサンプル

```
{
  "deviceMachineName": "FaceDevice1",
  "deviceAudioVolume": 98,
  "deviceScreenBrightness": 43,
  "accessType": 4,
  ....<略>
}
```

レスポンス仕様

デバイス基本設定		
項目	型	概要
deviceMachineName	文字列	デバイス名
deviceAudioVolume	数値	案内音声のボリューム(0-100)
deviceScreenBrightness	数値	画面の明るさ(0-100)
deviceLedBrightness	数値	LED 照明の明るさ(0-100)
deviceStandbyTime	数値	スタンバイに入る時間(0~86400)秒
deviceWorkstateTime	数値	スクリーンセーバーに入る時間(-1 ~ 86400)秒
アクセス制御設定		
項目	型	概要
accessType	数値	アクセスタイプの設定状況。 1: 顔認識, 3: 顔認識+カード, 4: 顔認識/カード
ネットワーク設定		

項目	型	概要
netDefaultNetCard	文字列	LAN カード(参照のみ。eth0 固定。)
netEth0MacAddress	文字列	MAC アドレス(参照のみ。)
netEth0Ipv4DhcpEnable	真偽値	IPv4 設定: DHCP を有効にするかどうか 1 or 0
netEth0Ipv4IpAddress	文字列	IPv4 設定: IP アドレス
netEth0Ipv4SubnetMask	文字列	IPv4 設定: サブネットマスク
netEth0Ipv4DefaultGateway	文字列	IPv4 設定: デフォルトゲートウェイ
netEth0Ipv4DnsServer1	文字列	IPv4 設定: 優先 DNS サーバー
netEth0Ipv4DnsServer2	文字列	IPv4 設定: 代替 DNS サーバー
netEth0Ipv6LinkLocalAddress	文字列	IPv6 設定: リンクアドレス
netEth0Ipv6IpAddress	文字列	IPv6 設定: IP アドレス
netEth0Ipv6IpAddressPrefix	文字列	IPv6 設定: IP アドレスプレフィックス
netEth0Ipv6DefaultGateway	文字列	IPv6 設定: デフォルトゲートウェイ
netEth0Ipv6DnsServer1	文字列	IPv6 設定: 優先 DNS サーバー
netEth0Ipv6DnsServer2	文字列	IPv6 設定: 代替 DNS サーバー
portHttp	数値	HTTP ポート
portRtsp	数値	RTSP ポート
portServer	数値	サーバーポート
ディスプレイ表示設定		
項目	型	概要
dispInfo	文字列	会社名などのカスタム表示。
dispShowName	真偽値	認識された人物の氏名の表示を行うかどうか 1 or 0
dispShowID	真偽値	認識された人物の ID の表示を行うかどうか 1 or 0
dispShowPhoto	真偽値	認識された人物の写真の表示を行うかどうか 1 or 0
dispShowIp	真偽値	デバイスの IP アドレスの表示を行うかどうか 1 or 0
dispShowSerialNo	真偽値	デバイスのシリアルナンバーの表示を行うかどうか 1 or 0
dispShowVersion	真偽値	デバイスのバージョン情報の表示を行うかどうか 1 or 0
dispShowPersonInfo	真偽値	デバイスの登録人数の表示を行うかどうか 1 or 0
dispShowOfflineData	真偽値	オフライン人数の表示を行うかどうか 1 or 0
通知表示設定		
項目	型	概要
tipsEnable	真偽値	認証成功時に通知表示するかどうか 1 or 0
tipsType	文字列 (列挙型)	認証成功時の通知表示 Welcome(登録者), RecognitOK(認証成功), PunchOK(カード認証成功), PassOK(通行許可),

		Custom(カスタム文字列)
tipsCustom	文字列	認証成功時の通知表示(カスタム文字列)
tipsBackgroundColor	文字列 (列挙型)	認証成功時のテキスト背景色 Blue / Red / Green
tipsVoiceEnable	真偽値	認証成功時に音声再生を行うかどうか 1 or 0
tipsVoiceType	文字列 (列挙型)	認証成功時の音声再生内容 Default, Welcome(登録者)のみ
strangerTipsEnable	真偽値	認証失敗時に通知表示するかどうか 1 or 0
strangerTipsType	文字列 (列挙型)	認証失敗時の通知表示 AccessFailNothit(未登録者), Custom(カスタム文字列)
strangerTipsCustom	文字列	認証失敗時の通知表示(カスタム文字列)
strangerTipsBackgroundColor	文字列 (列挙型)	認証失敗時のテキスト背景色 Blue / Red / Green
strangerVoiceEnable	真偽値	認証失敗時に音声再生を行うかどうか 1 or 0
strangerVoiceType	文字列 (列挙型)	認証失敗時の音声再生内容 現在利用可能は strangerVoice1(未登録者)のみ
識別設定		
項目	型	概要
recogWorkstateTime	数値	識別距離。0.5メートルから2メートルの範囲
recogLiveness	数値	識別レベル 1: 写真/ビデオの偽装を判別しない, 2: 写真/ビデオの偽装を部分的に判別する, 3: 写真/ビデオの偽装を正確に判別する
recogCircleInterval	数値	識別間隔(0-10)
recogSearchThreshold	数値	認識比較閾値(1-100)
recogMouthoccThreshold	数値	マスク検出時の認識比較閾値(1-100)
マスク検出設定		
項目	型	概要
maskDetectMode	数値	マスク検出の基本設定 0: マスク判定のみ, 1: マスクのない人をシールドする, 2: マスクのある人をシールドする
maskFaceAttrSwitch	数値	マスク検出モード 0: 口のみ覆うも許可する, 1: 鼻と口の両方を覆う
maskWearShowEnable	真偽値	マスクあり時に通知表示するかどうか 1 or 0
maskWearShowTips	文字列	マスクあり時の通知テキスト

maskWearShowBackgroundColor	文字列（列挙型）	マスクあり時の通知テキスト背景色 Blue / Green / Red
maskWearVoiceEnable	真偽値	マスクあり時に音声表示するかどうか 1 or 0
maskNowearShowEnable	真偽値	マスク無し時に通知表示するかどうか 1 or 0
maskNowearShowTips	文字列	マスク無し時の通知テキスト
maskNowearShowBackgroundColor	文字列（列挙型）	マスク無し時の通知テキスト背景色 Blue / Green / Red
maskNowearVoiceEnable	真偽値	マスク無し時に音声表示するかどうか 1 or 0
体温検出設定		
項目	型	概要
tempEnable	真偽値	体温検知の有効無効 1 or 0
tempDetectMode	数値	検知モード 0: 検出判定のみ, 1: 体温異常者をシールドする
tempUnit	数値	温度単位 0: 摂氏, 1: 華氏
tempValueRangeFrom	数値	正常体温の下限
tempValueRangeTo	数値	正常体温の上限
tempCorrection	数値	温度補正
tempLowTempCorrection	真偽値	低温補正を行うかどうか 1 / 0
tempMinPixel	数値	温度測定最小顔ピクセル 0-1024
tempNormalShowEnable	真偽値	体温正常時の通知表示をするか 1 / 0
tempNormalShowTips	文字列	体温正常時の通知表示する際の通知テキスト
tempNormalVoiceEnable	真偽値	体温正常時の音声通知をするか 1 / 0
tempAbnormalShowEnable	真偽値	体温異常時の通知表示をするか 1 / 0
tempAbnormalShowTips	文字列	体温異常時の通知表示する際の通知テキスト
tempAbnormalVoiceEnable	真偽値	体温異常時の音声通知をするか 1 / 0
NTP 設定		
項目	型	概要
ntpEnable	真偽値	NTP の有効無効 1 or 0
ntpHostName	文字列	NTP サーバーのホスト名
ntpPort	数値	NTP サーバーのポート
ntpInterval	数値	時刻同期間隔(分)

設定の更新

API URL	https://[HOST]/api1/config/setConfig
排他制御	有り (serialNo 単位に一リクエストのみを処理可能)

デバイスの設定情報を設定します。

この API を実行すると、即座にデバイスに反映処理を行うため応答までに時間を要する場合があります。

リクエストパラメータ

項目	必須		概要
serialNo	○	文字列	デバイスのシリアル No です。
デバイス基本設定			
項目	必須	型	概要
deviceMachineName		文字列	デバイス名(最大 32 文字)
deviceAudioVolume		数値	案内音声のボリューム(0-100)
deviceScreenBrightness		数値	画面の明るさ(0-100)
deviceLedBrightness		数値	LED 照明の明るさ(0-100)
deviceStandbyTime		数値	スタンバイに入る時間(0~86400)秒
deviceWorkstateTime		数値	スクリーンセーバーに入る時間(-1 ~ 86400)秒
アクセス制御設定			
項目	必須	型	概要
accessType		数値	アクセスタイプの設定状況。 1: 顔認識, 3: 顔認識+カード, 4: 顔認識/カード
ネットワーク設定			
項目	必須	型	概要
netEth0Ipv4DhcpEnable		真偽値	IPv4 設定: DHCP を有効にするかどうか 1 or 0
netEth0Ipv4IpAddress		文字列	IPv4 設定: IP アドレス(最大 15 文字)
netEth0Ipv4SubnetMask		文字列	IPv4 設定: サブネットマスク(最大 15 文字)
netEth0Ipv4DefaultGateway		文字列	IPv4 設定: デフォルトゲートウェイ(最大 15 文字)
netEth0Ipv4DnsServer1		文字列	IPv4 設定: 優先 DNS サーバー(最大 15 文字)
netEth0Ipv4DnsServer2		文字列	IPv4 設定: 代替 DNS サーバー(最大 15 文字)
netEth0Ipv6LinkLocalAddress		文字列	IPv6 設定: リンクアドレス(最大 39 文字)
netEth0Ipv6IpAddress		文字列	IPv6 設定: IP アドレス(最大 39 文字)

netEth0Ipv6IpAddressPrefix		文字列	IPv6 設定：IP アドレスプレフィックス(最大 39 文字)
netEth0Ipv6Defaultgateway		文字列	IPv6 設定：デフォルトゲートウェイ(最大 39 文字)
netEth0Ipv6DnsServer1		文字列	IPv6 設定：優先 DNS サーバー(最大 39 文字)
netEth0Ipv6DnsServer2		文字列	IPv6 設定：代替 DNS サーバー(最大 39 文字)
portHttp		数値	HTTP ポート (65535 までの数値)
portRtsp		数値	RTSP ポート (65535 までの数値)
portServer		数値	サーバーポート (65535 までの数値)
ディスプレイ表示設定			
項目	必須	型	概要
dispInfo		文字列	会社名などのカスタム表示。(最大 12 文字)
dispShowName		真偽値	認識された人物の氏名の表示を行うかどうか 1 or 0
dispShowID		真偽値	認識された人物の ID の表示を行うかどうか 1 or 0
dispShowPhoto		真偽値	認識された人物の写真の表示を行うかどうか 1 or 0
dispShowIp		真偽値	デバイスの IP アドレスの表示を行うかどうか 1 or 0
dispShowSerialNo		真偽値	デバイスのシリアルナンバーの表示を行うかどうか 1 or 0
dispShowVersion		真偽値	デバイスのバージョン情報の表示を行うかどうか 1 or 0
dispShowPersonInfo		真偽値	デバイスの登録人数の表示を行うかどうか 1 or 0
dispShowOfflineData		真偽値	オフライン人数の表示を行うかどうか 1 or 0
通知表示設定			
項目	必須	型	概要
tipsEnable		真偽値	認証成功時に通知表示するかどうか 1 or 0
tipsType		文字列(列挙型)	認証成功時の通知表示 Welcome(登録者), RecognitOK(認証成功), PunchOK(カード認証成功), PassOK(通行許可), Custom(カスタム文字列)
tipsCustom		文字列	認証成功時の通知表示(カスタム文字列)(最大 25 文字)
tipsBackgroundColor		文字列(列挙型)	認証成功時のテキスト背景色 Blue / Red / Green
tipsVoiceEnable		真偽値	認証成功時に音声再生を行うかどうか 1 or 0

tipsVoiceType		文字列(列挙型)	認証成功時の音声再生内容 現在利用可能は Default , Welcome(登録者) のみ
strangerTipsEnable		真偽値	認証失敗時に通知表示するかどうか 1 or 0
strangerTipsType		文字列(列挙型)	認証失敗時の通知表示 AccessFailNothit(未登録者), Custom(カスタム文字列)
strangerTipsCustom		文字列	認証失敗時の通知表示(カスタム文字列) (最大 25 文字)
strangerTipsBackgroundColor		文字列(列挙型)	認証失敗時のテキスト背景色 Blue / Red / Green
strangerVoiceEnable		真偽値	認証失敗時に音声再生を行うかどうか 1 or 0
strangerVoiceType		文字列(列挙型)	認証失敗時の音声再生内容 現在利用可能は strangerVoice1(未登録者) のみ
識別設定			
項目	必須	型	概要
recogWorkstateTime		数値	識別距離。0.5 メートルから 2 メートルの範囲(指定可能な値は 0.5 / 1 / 1.5 / 2 のいずれか。)
recogLiveness		数値	識別レベル 1: 写真/ビデオの偽装を判別しない, 2: 写真/ビデオの偽装を部分的に判別する, 3: 写真/ビデオの偽装を正確に判別する
recogCircleInterval		数値	識別間隔(0-10)
recogSearchThreshold		数値	認識比較閾値(1-100)
recogMouthoccThreshold		数値	マスク検出時の認識比較閾値(1-100)
マスク検出設定			
項目	必須	型	概要
maskDetectMode		数値	マスク検出の基本設定 0: マスク判定のみ, 1: マスクのない人をシールドする, 2: マスクのある人をシールドする
maskFaceAttrSwitch		数値	マスク検出モード 0: 口のみ覆うも許可する, 1: 鼻と口の両方を覆う
maskWearShowEnable		真偽値	マスクあり時に通知表示するかどうか 1 or 0
maskWearShowTips		文字列	マスクあり時の通知テキスト(最大 25 文字)
maskWearShowBackgroundColor		文字列(列挙型)	マスクあり時の通知テキスト背景色 Blue / Green / Red
maskWearVoiceEnable		真偽値	マスクあり時に音声表示するかどうか 1 or 0

maskNowearShowEnable		真偽値	マスク無し時に通知表示するかどうか 1 or 0
maskNowearShowTips		文字列	マスク無し時の通知テキスト(最大 25 文字)
maskNowearShowBackgroundColor		文字列(列挙型)	マスク無し時の通知テキスト背景色 Blue / Green / Red
maskNowearVoiceEnable		真偽値	マスク無し時に音声表示するかどうか 1 or 0
体温検出設定			
項目	必須	型	概要
tempEnable		真偽値	体温検知の有効無効 1 or 0
tempDetectMode		数値	検知モード 0: 検出判定のみ, 1: 体温異常者をシールドする
tempUnit		数値	温度単位 0: 摂氏, 1: 華氏
tempValueRangeFrom		数値	正常体温の下限(摂氏の場合は 34.0 - 42.0 華氏の場合は 93.2 - 107.6 小数点以下第一位まで)
tempValueRangeTo		数値	正常体温の上限(摂氏の場合は 34.0 - 42.0 華氏の場合は 93.2 - 107.6 小数点以下第一位まで。)
tempCorrection		数値	温度補正(-5 から 5 小数点以下第一位まで)
tempLowTempCorrection		真偽値	低温補正を行うかどうか 1 / 0
tempMinPixel		数値	温度測定最小顔ピクセル 0-1024
tempNormalShowEnable		真偽値	体温正常時の通知表示をするか 1 / 0
tempNormalShowTips		文字列	体温正常時の通知表示する際の通知テキスト(最大 25 文字)
tempNormalVoiceEnable		真偽値	体温正常時の音声通知をするか 1 / 0
tempAbnormalShowEnable		真偽値	体温異常時の通知表示をするか 1 / 0
tempAbnormalShowTips		文字列	体温異常時の通知表示する際の通知テキスト(最大 25 文字)
tempAbnormalVoiceEnable		真偽値	体温異常時の音声通知をするか 1 / 0
NTP 設定			
項目	必須	型	概要
ntpEnable		真偽値	NTP の有効無効 1 or 0
ntpHostName		文字列	NTP サーバーのホスト名(最大 100 文字)
ntpPort		数値	NTP サーバーのポート(0-65535)
ntpInterval		数値	時刻同期間隔(分)(1-1440) pass

レスポンスサンプル

```
{
  "updated":["deviceAudioVolume", "deviceScreenBr ightness"],
  "errored":["deviceStandbyTime", "accessType"]
}
```

レスポンス仕様

項目	型	概要
updated	配列(文字列)	設定の反映が正常に行われたパラメータ名です。既に設定済みの値と同じ値を指定した場合、そのパラメータに対して更新処理は行われません。(実際に更新が行われていない項目は、このレスポンスには含まれません。)
errored	配列(文字列)	設定の反映を行おうとしたものの、正常に行われなかったパラメータ名です。設定の反映処理の途中でデバイスとの接続が切れた場合などに発生する事があります。

待ち受けロゴ画像の取得

API URL	https://[HOST]/api1/config/getLogo
排他制御	無し

デバイスに設定されたロゴ画像を取得します。

この API を実行すると、その都度デバイスにアクセスし、最新の設定データを取得します。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN oです。

レスポンスサンプル(成功した場合)

<png 画像のバイナリが返却されます>

待ち受けロゴ画像の設定

API URL	https://[HOST]/api1/config/setLogo
排他制御	有り（serialNo 単位に一リクエストのみを処理可能）
Method	POST

デバイスの設定情報を設定します。

この API を実行すると、応答までに時間を要する場合があります。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN o です。
data	○	文字列	png 形式の画像ファイルを Base64 エンコードした文字列を指定して下さい。 Base64 エンコード後のファイルサイズが 5MB 以内である必要があります。

レスポンスサンプル

```
{  
  "result": true  
}
```

レスポンス仕様

項目	型	概要
result	真偽値	true: 待ち受けロゴ画像がデバイスに正常に反映された事を示します。

システム情報の取得

API URL	https://[HOST]/api1/system/getSystemInfo
排他制御	無し

デバイスのシステム情報を返却します。この情報は参照する事のみが可能です。
この API を実行すると、その都度デバイスにアクセスし、最新の設定データを取得します。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアル No です。

レスポンスサンプル

```
{
  "deviceType": "STD-XXX0721-X-XX05",
  "serialNumber": "XXXXXXXXXX",
  "softwareVersion": "1.396.11.10.T",
  "softwareBuildDate": "2020-08-19"
}
```

レスポンス仕様

項目	型	概要
deviceType	文字列	デバイスの型番です。
serialNumber	文字列	デバイスのシリアル No です。
softwareVersion	文字列	ソフトウェアのバージョン番号です。
softwareBuildDate	文字列	ソフトウェアのビルド日付です。YYYY-MM-DD 形式で返却されます。

デバイスの再起動（即座）

API URL	https://[HOST]/api1/system/reboot
排他制御	有り（serialNo 単位に一リクエストのみを処理可能）

デバイスを再起動します。

この機能を使うと、デバイスは即座に再起動処理を開始するため、デバイスからの成否についてのレスポンスは受け取る事は出来ません。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN oです。

レスポンスサンプル

```
{
  "request": true
}
```

レスポンス仕様

項目	型	概要
request	真偽値	入力チェックエラーや API トークンのエラーなどを除き、常に true を返却します。 このレスポンスはクラウドサーバーがデバイスに対して再起動要求を送った事のみを示すものであり、再起動の要求が正常に受け付けられた事を保証するものではありません。

デバイスのドア開錠(即座)

API URL	https://[HOST]/api1/system/openOnce
排他制御	無し

デバイスに接続されているドア(自動ドア、電子錠、ゲートなど)を強制的に開錠します。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN oです。

レスポンスサンプル

```
{
  "result": true
}
```

レスポンス仕様

項目	型	概要
result	真偽値	true: ドアが開錠されたことを示します。

デバイスへの休止指示

API URL	https://[HOST]/api1/system/setHibernateMode
排他制御	無し

デバイスを休止状態にします。休止状態のデバイスは、顔をかざしても顔認証が実行されず、画面状態(スクリーンセーバー、スタンバイ等)を維持します。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN oです。

レスポンスサンプル

```
{
  "result": true
}
```

レスポンス仕様

項目	型	概要
result	真偽値	true: デバイスが休止状態になったことを示します。

デバイスへの運用指示

API URL	https://[HOST]/api1/system/setOperationMode
排他制御	無し

デバイスを運用状態にします。デバイスが休止状態になっている場合、この API を使用することで顔認証可能な通常の状態に戻すことができます。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN o です。

レスポンスサンプル

```
{
  "result": true
}
```

レスポンス仕様

項目	型	概要
result	真偽値	true: デバイスが運用状態になったことを示します。

デバイス状態の確認（運用/休止）

API URL	https://[HOST]/api1/system/getCurrentMode
排他制御	無し

デバイスが運用状態か休止状態かを確認することができます。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN oです。

レスポンスサンプル

```
{
  "result": true,
  "state": 1
}
```

レスポンス仕様

項目	型	概要
result	真偽値	true: 状態の確認ができたことを示します。
state	0 もしくは 1	0:運用状態であることを示します。 1:が休止状態であることを示します。

デバイス画面へのメッセージ表示

API URL	https://[HOST]/api1/system/displayMessage
排他制御	無し

デバイス画面へ任意のメッセージを表示することができます。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN oです。
Customtip	○	文字列	メッセージ表示内容です。 25 文字以下での指定となります。
Tipstime	○	数値	メッセージ表示時間です。 ミリ秒での入力となります。
BorderColor	○	文字列※	メッセージ文字色です。
BackgroundColor	○	文字列※	メッセージ背景色です。

※半角カンマ区切りで RGBA の入力 (0~255) を行ってください。正しく入力できていない場合は、文字色は黒字、背景色は透明となります。

レスポンスサンプル

```
{
  "result": true
}
```

レスポンス仕様

項目	型	概要
result	真偽値	true: メッセージ表示ができたことを示します。

デバイスの自動起動設定を取得

API URL	https://[HOST]/api1/system/getRebootSchedule
排他制御	無し

デバイスの自動再起動設定を取得します。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN oです。

レスポンスサンプル

```
{
  "rebootScheduleEnable": 0,
  "rebootScheduleDayWeek": 0,
  "rebootScheduleHours": 5
}
```

レスポンス仕様

項目	型	概要
rebootScheduleEnable	真偽値	自動再起動が有効な場合は 1 となります。無効な場合は 0 です。
rebootScheduleDayWeek	数値	設定済みの自動再起動の曜日です。 0:日, 1:月, 2:火, 3:水, 4:木,5:金,6:土
rebootScheduleHours	数値	設定済みの自動再起動の時刻です。(0 時から 23 時)

デバイスの自動起動設定を更新

API URL	https://[HOST]/api1/system/setRebootSchedule
排他制御	有り (serialNo 単位に一リクエストのみを処理可能)

デバイスの自動再起動の設定情報を更新します。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN oです。
rebootScheduleEnable		数値	自動再起動が有効とする場合は1 となります。無効とする場合は0 です。
rebootScheduleDayWeek		数値	自動再起動を設定する曜日です。 0:日, 1:月, 2:火, 3:水, 4:木,5:金,6:土
rebootScheduleHours		数値	自動再起動を設定する時刻です。(0 時から 23 時)

レスポンスサンプル

```
{
  "updated": ["rebootScheduleDayWeek", "rebootScheduleHours" ],
  "errored": []
}
```

レスポンス仕様

項目	型	概要
updated	配列(文字列)	設定の反映が正常に行われたパラメータ名です。既に設定済みの値と同じ値を指定した場合、そのパラメータに対して更新処理は行われません。(実際に更新が行われていない項目は、このレスポンスには含まれません。)
errored	配列(文字列)	設定の反映を行おうとしたものの、正常に行われなかったパラメータ名です。設定の反映処理の途中でデバイスとの接続が切れた場合などに発生する事があります。

設定のエクスポート

API URL	https://[HOST]/api1/system/exportConfig
排他制御	無し

デバイスに設定された設定一式をファイル化し、取得します。

この API を実行すると、その都度デバイスにアクセスし、最新の設定データを取得します。

ここで取得出来るファイルはバイナリ形式のため、テキストエディタで編集出来るものではありません。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアルN oです。

レスポンスサンプル

<設定ファイルのバイナリが返却されます>

設定のインポート

AP	
排他	
Me	
設定	
この	ルを
指定	利用
くだ	
リク	
項目	
ser	
dat	ト
	バイ
	ード
	ノコ
	ある
この機能は現在ご利用いただけません。	
レス	
{	
}	
レス	
項目	
req	を除
	付し
	バイ
	スに要求が正常に受け付けられた事を保証するものでは ありません。

ファームウェアアップデート

API URL	https://[HOST]/api1/system/updateFirmware
排他制御	有り（serialNo 単位に一リクエストのみを処理可能）

デバイスのファームウェアを更新します。

この機能を使うと、デバイスの再起動が掛かります。また、もしデバイスにしか格納されていない人物データなどが存在している場合には、この機能を頂く前にはバックアップを取っておいて頂くようお願いを致します。（クラウドサーバーに保存されたデータには影響はありません。）

ファームウェアファイルはクラウドサーバーに保存されており、デバイスはクラウドサーバーからファイルをダウンロードする事で、アップデートを実行します。そのため、アップデート中はデバイスのインターネット接続を維持して下さい。

また、この機能を使うと、デバイスは即座にファームウェアアップデート処理を開始するため、デバイスからの成否についてのレスポンスは受け取る事は出来ません。

処理結果を確認するためには、数分待ってから、システム情報の取得 (/system/getSystemInfo) を実行し、結果を確認して下さい。

リクエストパラメータ

項目	必須	型	概要
serialNo	○	文字列	デバイスのシリアル No です。
version	○	文字列	ファームウェアのバージョン名を指定します。

レスポンスサンプル

```
{
  "request": true
}
```

レスポンス仕様

項目	型	概要
request	真偽値	入力チェックエラーや API トークンのエラーなどを除き、常に true を返却します。 レスポンスはクラウドサーバーがデバイスに対して再起動要求を送った事のみを示すものであり、デバイスに対して要求が正常に受け付けられた事を保証するものではありません。

