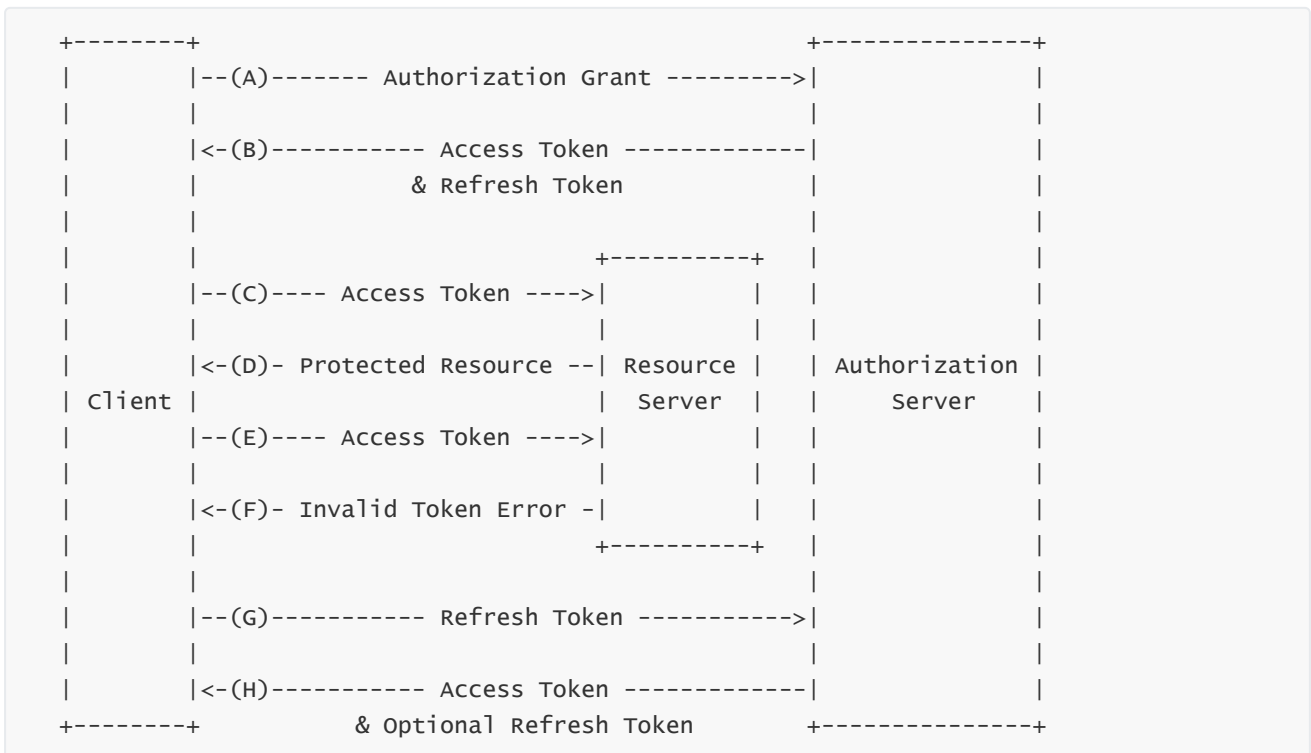


# Trimlight V2 Oauth API Documentation

| Version | Description  | Date       |
|---------|--|------------|
| 1.0.0   | initial version.   | 2021-04-20 |
| 1.1.0   | Add <code>currentEffect</code> data to the device details response data. | 2021-11-22 |
| 1.2.0   | Add group function APIs. (19 - 23)                                       | 2021-12-04 |
| 1.2.1   | Add the "sync a group" API. (24)   | 2021-12-08 |
| 1.3.0   | Add the "Notify update shadow data" API. (25)                            | 2022-08-18 |

## 1. Oauth2



The flow illustrated in Figure 2 includes the following steps:

- (A) The client requests an access token by authenticating with the authorization server and presenting an authorization grant.

- (B) The authorization server authenticates the client and validates the authorization grant, and if valid, issues an access token and a refresh token.
- (C) The client makes a protected resource request to the resource server by presenting the access token.
- (D) The resource server validates the access token, and if valid, serves the request.
- (E) Steps (C) and (D) repeat until the access token expires. If the client knows the access token expired, it skips to step (G); otherwise, it makes another protected resource request.
- (F) Since the access token is invalid, the resource server returns an invalid token error.
- (G) The client requests a new access token by authenticating with the authorization server and presenting the refresh token. The client authentication requirements are based on the client type and on the authorization server policies.
- (H) The authorization server authenticates the client and validates the refresh token, and if valid, issues a new access token (and, optionally, a new refresh token).

## 1.1 Grant URL

```
GET https://trimlight.ledhue.com/trimlight/v1/oauth/grant?client_id=
<clientId>&redirect_uri=<redirectUri>&scope=<scope>&state=<state>
```

## 1.2 Get access token

```
POST https://trimlight.ledhue.com/trimlight/v1/oauth/accessToken
```

```
Content-Type: application/x-www-form-urlencoded
```

```
code=<grant_code>&grant_type=authorization_code&client_id=<clientId>&client_secret=
<clientSecret>
```

**Please contact our business to obtain [clientId] and [clientSecret]**

Response:

```
{
  "access_token": "23448590d9c43d230dd7f8b5e7876923",
  "token_type": "Bearer",
  "expires_in": 7200, // access token expiration time (seconds)
  "refresh_token": "a7de0c6b127df94e6379924b076d04ea"
}
```

Refresh token:

```
POST https://trimlight.ledhue.com/trimlight/v1/oauth/accessToken

Content-Type: application/x-www-form-urlencoded

code=<grant_code>&grant_type=refresh_token&client_id=<clientId>&client_secret=
<clientSecret>&refresh_token=<refreshToken>
```

Response:

```
{
  "access_token": "23448590d9c43d230dd7f8b5e7876923",
  "token_type": "Bearer",
  "expires_in": 7200,
  "refresh_token": "a7de0c6b127df94e6379924b076d04ea"
}
```

## 2. Base URL

---

```
POST https://trimlight.ledhue.com/trimlight
```

**PS: All the following request must add the http request header:**

key: **authorization**

value:

## 3. Get device list

---

```
GET /v1/oauth/resources/devices
```

**Request params:**

```
{
  "page": 1 // 10 devices on one page
           // If the value is 0 or null, it will return to the list of all devices
}
```

### Response result:

```
{
  "code": 0,
  "desc": "success",
  "payload": {
    "total": 2,
    "current": 1,
    "data": [
      {
        "deviceId": "xxxxxxxxxx1",
        "name": "xxx1",
        "switchState": 0,
        "connectivity": 1,
        "state": 0,
        "fwVersionName": "1.1.1"
      },
      {
        "deviceId": "xxxxxxxxxx2",
        "name": "xxx2",
        "switchState": 0,
        "connectivity": 1,
        "state": 0,
        "fwVersionName": "1.1.1"
      }
    ]
  }
}
```

### Response result code details

| Field   | Description        | Type    |
|---------|--------------------|---------|
| code    | Result code. [1]   | Integer |
| desc    | Result decription. | String  |
| payload | Result payload.    | Object  |

### Page fields details

| Field | Description                          | Type    |
|-------|--------------------------------------|---------|
| total | Number of all devices.               | Integer |
| page  | The page number of the current data. | Integer |
| data  | device list.                         | List    |

#### Device fields details

| Field         | Description   | Type    |
|---------------|---|---------|
| deviceId      | Unique ID of the device.  | String  |
| name          | Device name.  | String  |
| switchState   | Device switch state.<br>0 : light off.<br>1 : manual mode.<br>2 : timer mode. | Integer |
| connectivity  | Device connectivity state.<br>0 : offline.<br>1 : online.                     | Integer |
| state         | Device state.<br>0 : normal.<br>1 : upgrading.                                | Integer |
| fwVersionName | Device firmware version name.   | String  |

## 4. Get device detail data

POST /v1/oauth/resources/device/get

Request params:

```

{
  "deviceId": "<device-id>",
  "currentDate": {
    "year": 21, // 2021
    "month": 1,
    "day": 1,
    "weekday": 1,
    "hours": 1,
    "minutes": 1,
    "seconds": 1
  }
}

```

### Current date details

| Field   | Description   | Type    |
|---------|---|---------|
| year    | Current date year. <b>Years relative to 2000.</b>   | Integer |
| month   | Current date month.<br><b>Range: [1, 12].</b>   | Integer |
| day     | Current date day.<br><b>Range: [1, 31].</b>   | Integer |
| weekday | Current day of week.<br><b>SUNDAY = 1,</b><br><b>MONDAY = 2,</b><br><b>TUESDAY = 3,</b><br><b>WEDNESDAY = 4,</b><br><b>THURSDAY = 5,</b><br><b>FRIDAY = 6,</b><br><b>SATURDAY = 7</b> | Integer |
| hours   | Current date year.<br><b>Range: [0, 23].</b>  | Integer |
| minutes | Current date year.<br><b>Range: [0, 59].</b>  | Integer |
| seconds | Current date year.<br><b>Range: [0, 59].</b>  | Integer |

### Response result:

```

{
  "code": 0,
  "desc": "success",
  "payload": {
    "name": "xxxx2",

```

```
"switchState": 0,
"connectivity": 1,
"state": 0,
"colorOrder": 0,
"ic": 0,
"ports": [
  {
    "id": 0,
    "start": 1,
    "end": 1024
  },
  {
    "id": 1,
    "start": 1,
    "end": 1024
  },
  {
    "id": 2,
    "start": 1,
    "end": 1024
  },
  {
    "id": 3,
    "start": 1,
    "end": 1024
  }
],
"fwVersionName": "1.1.1",
"effects": [
  {
    "id": 0,
    "name": "New Year",
    "category": 0,
    "mode": 0,
    "speed": 100,
    "brightness": 100,
    "pixelLen": 30,
    "reverse": false
  },
  {
    "id": 1,
    "name": "xxxxxxxxx2",
    "category": 1,
    "mode": 0,
    "speed": 100,
    "brightness": 100,
    "pixels": [
      {
        "index": 0,
        "count": 5,
        "color": 16711680,
        "disable": false
      }
    ]
  },

```

```
        {
            "index": 1,
            "count": 10,
            "color": 65280,
            "disable": false
        },
        {
            "index": 2,
            "count": 10,
            "color": 255,
            "disable": false
        }
    ]
},
"combinedEffect": {
    "effectIds": [0, 2, 3],
    "interval": 5
},
"daily": [
    {
        "id": 0,
        "enable": true,
        "effectId": 0,
        "repetition": 1,
        "startTime": {
            "hours": 10,
            "minutes": 1
        },
        "endTime": {
            "hours": 11,
            "minutes": 1
        }
    },
    {
        "id": 1,
        "enable": true,
        "effectId": 1,
        "repetition": 1,
        "startTime": {
            "hours": 10,
            "minutes": 1
        },
        "endTime": {
            "hours": 11,
            "minutes": 1
        }
    }
],
"calendar": [
    {
        "id": 0,
        "effectId": 1,
```



```

        "startDate": {
            "month": 12,
            "day": 31
        },
        "endDate": {
            "month": 1,
            "day": 1
        },
        "startTime": {
            "hours": 10,
            "minutes": 1
        },
        "endTime": {
            "hours": 11,
            "minutes": 1
        }
    },
    {
        "id": 1,
        "effectId": 2,
        "startDate": {
            "month": 12,
            "day": 31
        },
        "endDate": {
            "month": 1,
            "day": 1
        },
        "startTime": {
            "hours": 10,
            "minutes": 1
        },
        "endTime": {
            "hours": 11,
            "minutes": 1
        }
    }
],
"currentEffect": {
    "category": 1,
    "mode": 1,
    "speed": 174,
    "brightness": 204,
    "pixelLen": 37,
    "reverse": false
}
}

```

## 4.1 Device fields details

| Field                       | Description  | Type    |
|-----------------------------|--|---------|
| <code>name</code>           | Device name.   | String  |
| <code>switchState</code>    | Device switch state.<br>0 : light off.<br>1 : manual mode.<br>2 : timer mode.  | Integer |
| <code>connectivity</code>   | Device connectivity state.<br>0 : offline.<br>1 : online.  | Integer |
| <code>state</code>          | Device state.<br>0 : normal.<br>1 : upgrading.   | Integer |
| <code>colorOrder</code>     | Color order. <b>[2]</b> .  | Integer |
| <code>ic</code>             | IC. <b>[3]</b>   | Integer |
| <code>ports</code>          | The pixel setting for each port.<br><b>id: Port ID, range from 0 to 3, correspond to port1, port2, port3, port4.</b><br><b>start: The start pixel of the port. range: [1, 2048]</b><br><b>end: The end pixel of the port. range: [1, 2048].</b><br><b>"start" should not greater than "end".</b> | List    |
| <code>fwVersionName</code>  | Device firmware version name.  | String  |
| <code>effects</code>        | All the effects stored in the device.  | List    |
| <code>combinedEffect</code> | Combine effect.  | Object  |
| <code>daily</code>          | Daily schedules.<br>Each device has two daily schedules.   | List    |
| <code>calendar</code>       | Calendar schedules.  | List    |
| <code>currentEffect</code>  | Device current running effect.<br>(If the device's switch state is timer mode, although the light is off at this time, it will return the last running effect data.)<br><b>Note: The effect ID will be -1, when the controller is running a preview effect (not yet saved).</b>                  | Object  |

## 4.2 Effect fields details

| Field                   | Description   | Type    |
|-------------------------|---|---------|
| <code>id</code>         | Effect ID.<br>ID of the saved effect.<br><b>Note that it may be -1 in <code>currentEffect</code>, when the controller is running a preview effect (not yet saved).</b>                        | Integer |
| <code>category</code>   | Effect category.<br>0 : build-in effect.<br>1 : custom effect.  | Integer |
| <code>mode</code>       | Effect mode.<br><b>Build-in effect (category value is 0) mode range: [0, 179].</b> <a href="#">[4]</a><br><b>Custom effect (category value is 1) mode range: [0, 16].</b> <a href="#">[5]</a> | Integer |
| <code>speed</code>      | Effect speed.<br><b>Speed range: [0, 255].</b>  | Integer |
| <code>brightness</code> | Effect brightness.<br><b>Brightness range: [0, 255].</b>  | Integer |
| <code>pixelLen</code>   | Effect pixel length.<br><b>(Only required for build-in effects)</b><br><b>Pixel length range: [1, 90].</b>  | Integer |
| <code>reverse</code>    | Reverse effect.<br><b>(Only required for build-in effects)</b>  | Boolean |
| <code>pixels</code>     | Custom effect pixels.<br><b>(Only required for build-in effects)</b>  | List    |

### 4.3 Pixel fields details

| Field                | Description                | Type    |
|----------------------|----------------------------|---------|
| <code>index</code>   | Custom effect pixel index. | Integer |
| <code>count</code>   | Custom effect pixel count. | Integer |
| <code>color</code>   | Custom effect pixel color. | Integer |
| <code>disable</code> | Disable pixel.             | Boolean |

### 4.4 Combined effect fields details

| Field                  | Description   | Type    |
|------------------------|---|---------|
| <code>effectIds</code> | A list of each effects' ID in the combined effect.                        | List    |
| <code>interval</code>  | The interval between switching to the next effect. <b>(Unit: minute.)</b> | Integer |

## 4.5 Daily schedule fields details

| Field                   | Description   | Type    |
|-------------------------|---|---------|
| <code>id</code>         | Daily schedule ID.<br><b>0: daily schedule 1.</b><br><b>1: daily schedule 2.</b>    | List    |
| <code>enable</code>     | Enable daily schedule.  | Boolean |
| <code>effectId</code>   | Effect ID in the daily schedule.  | Integer |
| <code>repetition</code> | The repetition of daily schedule.   | Integer |
| <code>startTime</code>  | The start time of daily schedule. <a href="#">See schedule time fields details.</a> | Object  |
| <code>endTime</code>    | The end time of daily schedule. <a href="#">See schedule time fields details.</a>   | Object  |

## 4.6 Schedule Time fields details

| Field                | Description                           | Type    |
|----------------------|---------------------------------------|---------|
| <code>hours</code>   | Schedule hours, range from [1, 24].   | Integer |
| <code>minutes</code> | Schedule minutes, range from [1, 60]. | Integer |

## 4.7 Calendar schedule fields details

| Field                  | Description   | Type    |
|------------------------|---|---------|
| <code>id</code>        | Calendar schedule ID.<br><b>Schedule ID range: [0, 59].</b><br><b>Up to 60 calendar schedules can be saved.</b> | List    |
| <code>effectId</code>  | Effect ID in the calendar schedule.   | Integer |
| <code>startDate</code> | The start date of calendar schedule. <a href="#">See schedule date fields details.</a>                          | Object  |
| <code>endDate</code>   | The end date of calendar schedule. <a href="#">See schedule date fields details.</a>                            | Object  |
| <code>startTime</code> | The start time in each day of calendar schedule. <a href="#">See schedule time fields details.</a>              | Object  |
| <code>endTime</code>   | The end time in each day of calendar schedule. <a href="#">See schedule time fields details.</a>                | Object  |

## 4.8 Schedule date fields details

| Field              | Description                         | Type    |
|--------------------|-------------------------------------|---------|
| <code>month</code> | Schedule month, range from [1, 12]. | Integer |
| <code>day</code>   | Schedule day, range from [1, 31].   | Integer |

## 4.9 Effect fields details

| Field                   | Description   | Type    |
|-------------------------|---|---------|
| <code>category</code>   | Effect category.<br>0 : build-in effect.<br>1 : custom effect.  | Integer |
| <code>mode</code>       | Effect mode.<br><b>Build-in effect (category value is 0) mode range: [0, 179].</b> <a href="#">[4]</a><br><b>Custom effect (category value is 1) mode range: [0, 16].</b> <a href="#">[5]</a> | Integer |
| <code>speed</code>      | Effect speed.<br><b>Speed range: [0, 255].</b>  | Integer |
| <code>brightness</code> | Effect brightness.<br><b>Brightness range: [0, 255].</b>  | Integer |
| <code>pixelLen</code>   | Effect pixel length.<br><b>(Only available for build-in effects)</b><br><b>Pixel length range: [1, 90].</b>   | Integer |
| <code>reverse</code>    | Reverse effect.<br><b>(Only available for build-in effects)</b>   | Boolean |
| <code>pixels</code>     | Custom effect pixels.<br><b>(Only available for build-in effects)</b>   | List    |

## 5. Set device switch state

**POST** `/v1/oauth/resources/device/update`

### Request params:

```
{
  "deviceId": "<device-id>",
  "payload": {
    "switchState": 0
  }
}
```

### Response result:

```
{
  "code": 0,
  "desc": "success"
}
```

## 6. Set device name

---

```
POST /v1/oauth/resources/device/update
```

### Request params:

```
{
  "deviceId": "<device-id>",
  "payload": {
    "name": "xxx"
  }
}
```

### Response result:

```
{
  "code": 0,
  "desc": "success"
}
```

## 7. Set device color order

---

```
POST /v1/oauth/resources/device/update
```

### Request params:

```
{
  "deviceId": "<device-id>",
  "payload": {
    "colorOrder": 0
  }
}
```

### Response result:

```
{
  "code": 0,
  "desc": "success"
}
```

## 8. Set device IC

---

```
POST /v1/oauth/resources/device/update
```

### Request params:

```
{
  "deviceId": "<device-id>",
  "payload": {
    "ic": 0
  }
}
```

### Response result:

```
{
  "code": 0,
  "desc": "success"
}
```

## 9. Set device port

---

```
POST /v1/oauth/resources/device/update
```

### Request params:

```
{
  "deviceId": "<device-id>",
  "payload": {
    "ports": [
      {
        "id": 0, // [0, 3]
        "start": 1,
        "end": 1024
      }
    ]
  }
}
```

### Response result:

```
{
  "code": 0,
  "desc": "success"
}
```

## 10. Preview build-in effect

```
POST /v1/oauth/resources/device/effect/preview
```

### Request params:

```
{
  "deviceId": "<device-id>",
  "payload": {
    "category": 0,
    "mode": 0,
    "speed": 100,
    "brightness": 100,
    "pixelLen": 10,
    "reverse": false
  }
}
```

### Effect fields details

| Field      | Description   | Type    |
|------------|---|---------|
| category   | Effect category.<br>0 : build-in effect.<br>1 : custom effect.<br><b>Here the value is 0.</b> | Integer |
| mode       | Build-in effect mode.<br><b>Build-in effect mode range: [0, 179].</b> <a href="#">[4]</a>     | Integer |
| speed      | Effect speed.<br><b>Speed range: [0, 255].</b>  | Integer |
| brightness | Effect brightness.<br><b>Brightness range: [0, 255].</b>                                      | Integer |
| pixelLen   | Effect pixel length.<br><b>Pixel length range: [1, 90].</b>                                   | Integer |
| reverse    | Reverse effect.   | Boolean |



## Response result:

```
{
  "code": 0,
  "desc": "success"
}
```

## Example

### Build-in effect

```
{
  "deviceId": "<device-id>",
  "payload": {
    "category": 1,
    "mode": 1,
    "speed": 100,
    "brightness": 100,
    "pixelLen": 10,
    "reverse": false
  }
}
```

# 11. Preview custom effect

```
POST /v1/oauth/resources/device/effect/preview
```

## Request params:

```
{
  "deviceId": "<device-id>",
  "payload": {
    "category": 1,
    "mode": 0,
    "speed": 100,
    "brightness": 100,
    "pixels": [
      {
        "index": 0,
        "count": 5,
        "color": 16711680, // (0xFF0000)
        "disable": false
      },
      {
        "index": 1,
        "count": 10,
        "color": 16711680,

```

```

    "disable": false
  }
]
}
}

```

### Effect fields details

| Field      | Description   | Type    |
|------------|---|---------|
| category   | Effect category.<br>0 : build-in effect.<br>1 : custom effect.<br><b>Here the value is 1.</b> | Integer |
| mode       | Effect mode.<br><b>Custom effect mode range: [0, 16].</b> <a href="#">[5]</a>                 | Integer |
| speed      | Effect speed.<br><b>Speed range: [0, 255].</b>  | Integer |
| brightness | Effect brightness.<br><b>Brightness range: [0, 255].</b>                                      | Integer |
| pixels     | Custom effect pixels.   | List    |

### Pixel fields details

| Field   | Description  | Type    |
|---------|--|---------|
| index   | Custom pixel index.<br><b>Index range: [0, 29].</b>            | Integer |
| count   | Custom pixel count.<br><b>Pixel count range: [0, 60].</b>      | Integer |
| color   | Custom pixel color decimal value.<br>eg: 0xFF0000 => 16711680. | Integer |
| disable | Custom pixel disable.  | Boolean |

### Example

#### Custom effect

```

{
  "deviceId": "<device-id>",
  "payload": {
    "category": 2,
    "mode": 1,

```

```

    "speed": 100,
    "brightness": 100,
    "pixels": [
      {
        "index": 0,
        "count": 10,
        "color": 255,
        "disable": false
      },
      {
        "index": 1,
        "count": 2,
        "color": 0,
        "disable": true
      },
      {
        "index": 2,
        "count": 3,
        "color": 65280,
        "disable": false
      },
      {
        "index": 2, // The same index will be overwritten
        "count": 3,
        "color": 16711680,
        "disable": false
      }
    ]
  }
}

```

## 12. Add/Update effect

POST [/v1/oauth/resources/device/effect/save](#)

### Request params:

```

{
  "deviceId": "<device-id>",
  "payload": {
    "id": -1
    "name": "xxxx",
    "category": 1/2,
    "mode": 0,
    "speed": 100,
    "brightness": 100,
    "pixelLen": 10,
    "reverse": false,
    "pixels": [

```

```

    {
      "index": 0,
      "count": 5,
      "color": 16711680, // (0xFF0000)
      "disable": false
    },
    {
      "index": 1,
      "count": 10,
      "color": 16711680,
      "disable": false
    }
  ]
}

```

### Effect fields details

| Field                   | Description  | Type    |
|-------------------------|--|---------|
| <code>id</code>         | Effect ID.<br><b>If ID is -1 or null, it will be saved as a new effect, if not, the effect corresponding to the ID will be updated.</b><br><b>Up to 60 effects can be saved.</b> | Integer |
| <code>category</code>   | Effect category.<br>0 : build-in effect.<br>1 : custom effect.   | Integer |
| <code>mode</code>       | Effect mode.<br><b>Custom effect mode range: [0, 16]. <a href="#">[5]</a></b>  | Integer |
| <code>speed</code>      | Effect speed.<br><b>Speed range: [0, 255].</b>   | Integer |
| <code>brightness</code> | Effect brightness.<br><b>Brightness range: [0, 255].</b>   | Integer |
| <code>pixelLen</code>   | Effect pixel length.<br><b>Pixel length range: [1, 90].</b><br><b>(Only required for build-in effects)</b>   | Integer |
| <code>reverse</code>    | Reverse effect.<br><b>(Only required for build-in effects)</b>   | Boolean |
| <code>pixels</code>     | Custom effect pixels. <a href="#">See pixel fields details.</a><br><b>(Only required for custom effects)</b>   | List    |

**Response result:**

```
{
  "code": 0,
  "desc": "success",
  "payload": {
    "id": 10 // effect id
  }
}
```

## 13. Check out effect

---

```
POST /v1/oauth/resources/device/effect/view
```

### Request params:

```
{
  "deviceId": "<device-id>",
  "payload": {
    "id": 0
  }
}
```

### Response result:

```
{
  "code": 0,
  "desc": "success"
}
```

## 14. Delete effect

---

```
POST /v1/oauth/resources/device/effect/delete
```

### Request prams:

```
{
  "deviceId": "<device-id>",
  "payload": {
    "id": 0
  }
}
```

### Response result:

```
{
  "code": 0,
  "desc": "success"
}
```

## 15. Update daily schedule

```
POST /v1/oauth/resources/device/daily/save
```

### Request prams:

```
{
  "deviceId": "<device-id>",
  "payload": {
    "id": 1,
    "enable": true,
    "effectId": 1,
    "repetition": 1,
    "startTime": {
      "hours": 10,
      "minutes": 1
    },
    "endTime": {
      "hours": 11,
      "minutes": 1
    },
    "currentDate": {
      "month": 1,
      "day": 1
    }
  }
}
```

### Response result:

```
{
  "code": 0,
  "desc": "success"
}
```

## 16. Add/Update calendar schedule

```
POST /v1/oauth/resources/device/calendar/save
```

### Request prams:

```
{
  "deviceId": "<device-id>",
  "payload": {
    "id": 0,
    "effectId": 1, // The combined effect id is fixed at 200
    "startDate": {
      "month": 12,
      "day": 31
    },
    "endDate": {
      "month": 1,
      "day": 1
    },
    "startTime": {
      "hours": 10,
      "minutes": 1
    },
    "endTime": {
      "hours": 11,
      "minutes": 1
    }
  }
}
```

#### Response result:

```
{
  "code": 0,
  "desc": "success",
  "payload": {
    "id": 10 // calendar id
  }
}
```

## 17. Delete calendar schedule

```
POST /v1/oauth/resources/device/calendar/delete
```

#### Request prams:

```
{
  "deviceId": "<device-id>",
  "payload": {
    "id": 0
  }
}
```

#### Response result:

```
{
  "code": 0,
  "desc": "success"
}
```

## 18. Set combined effect

```
POST /v1/oauth/resources/device/combined-effect/save
```

### Request prams:

```
{
  "deviceId": "<device-id>",
  "payload": {
    "effectIds": [0, 2, 3], // up to 5 effects
    "interval": 5
  }
}
```

### Response result:

```
{
  "code": 0,
  "desc": "success"
}
```

## 19. Get group list

```
GET /v1/oauth/resources/groups
```

### Request params:

```
{
  "page": 1 // 10 groups on one page
            // If the value is 0 or null, it will return to the list of all groups
}
```

### Response result:

```
{
  "code": 0,
  "desc": "success",
  "payload": {
    "total": 2,
  }
}
```



```

"current": 1,
"data": [
  {
    "groupId": "xxx",
    "name": "group1",
    "masterDevice": {
      "deviceId": "<device_id>",
      "name": "<device_name>"
    },
    "devices": [
      {
        "deviceId": "<device_id>",
        "name": "<device_name>"
      }
    ]
  },
  {
    "groupId": "xxx",
    "name": "group2",
    "masterDevice": {
      "deviceId": "<device_id>",
      "name": "<device_name>"
    },
    "devices": [
      {
        "deviceId": "<device_id>",
        "name": "<device_name>"
      }
    ]
  }
]
}

```

## 20. Add a new group

GET </v1/oauth/resources/group/add>

### Request params:

```

{
  "name": "<new group name>",
  "masterDevice": "<master device ID>",
  "devices": [
    "<master device ID>",
    "<device ID>",
    "<device ID>"
  ]
}

```

### Response result:

```
{
  "code": 0,
  "desc": "success",
  "payload": {
    "groupId": "<new group ID>"
  }
}
```

## 21. Update a group

---

```
GET /v1/oauth/resources/group/update
```

### Request params:

```
{
  "groupId": "<group ID>",
  "masterDevice": "<master device ID>",
  "devices": [
    "<master device ID>",
    "<device ID>",
    "<device ID>"
  ]
}
```

### Response result:

```
{
  "code": 0,
  "desc": "success",
  "payload": {
    "groupId": "<new group ID>"
  }
}
```

## 22. Rename a group

---

```
GET /v1/oauth/resources/group/rename
```

### Request params:

```
{
  "groupId": "<group ID>",
  "name": "<new group name>"
}
```

**Response result:**

```
{
  "code": 0,
  "desc": "success"
}
```

## 23. Delete a group

---

```
GET /v1/oauth/resources/group/delete
```

**Request params:**

```
{
  "groupId": "<group ID>"
}
```

**Response result:**

```
{
  "code": 0,
  "desc": "success"
}
```

## 24. Sync a group

---

```
GET /v1/oauth/resources/group/sync
```

**Request params:**

```
{
  "groupId": "<group ID>"
}
```

**Response result:**

```
{
  "code": 0,
  "desc": "success"
}
```

**PS: The master device must be online.**

## 25. Notify update shadow data

---

Before requesting detailed data for the device, you can send this request to notify the device to report the latest shadow data.

```
GET /v1/oauth/resources/device/notify-update-shadow
```

### Request params:

```
{
  "deviceId": "<device-id>",
  "currentDate": {
    "year": 21, // 2021
    "month": 1,
    "day": 1,
    "weekday": 1,
    "hours": 1,
    "minutes": 1,
    "seconds": 1
  }
}
```

### Response result:

```
{
  "code": 0,
  "desc": "success"
}
```

## Appendix

---

## [1] Result code

| Code  | Description     |
|-------|-----------------|
| 0     | Success.        |
| 10001 | Error.          |
| 10002 | Wrong password. |

## [2] Color order

| Color order | Value |
|-------------|-------|
| RGB         | 0     |
| RBG         | 1     |
| GRB         | 2     |
| GBR         | 3     |
| BRG         | 4     |
| BGR         | 5     |

## [3] IC

| IC      | Value |
|---------|-------|
| UCS1903 | 0     |
| DMX512  | 1     |

## [4] Build-in effect mode

| Color order           | Value |
|-----------------------|-------|
| Rainbow Gradual Chase | 0     |
| Rainbow Comet         | 1     |
| Rainbow Segment       | 2     |
| Rainbow Wave          | 3     |
| Rainbow Meteor        | 4     |
| Rainbow Gradual       | 5     |
| Rainbow Jump          | 6     |
| Rainbow Stars         | 7     |
| Rainbow Fade In Out   | 8     |
| Rainbow Spin          | 9     |
| Red Stacking          | 10    |
| Green Stacking        | 11    |
| Blue Stacking         | 12    |
| Yellow Stacking       | 13    |
| Cyan Stacking         | 14    |
| Purple Stacking       | 15    |
| White Stacking        | 16    |
| Full Color Stack      | 17    |
| Red to Green Stack    | 18    |
| Green to Blue Stack   | 19    |
| Blue to Yellow Stack  | 20    |
| Yellow to Cyan Stack  | 21    |
| Cyan to Purple Stack  | 22    |
| Purple to White Stack | 23    |
| Red Comet             | 24    |
| Green Comet           | 25    |
| Blue Comet            | 26    |

| Color order       | Value |
|-------------------|-------|
| Yellow Comet      | 27    |
| Cyan Comet        | 28    |
| Purple Comet      | 29    |
| White Comet       | 30    |
| Red Meteor        | 31    |
| Green Meteor      | 32    |
| Blue Meteor       | 33    |
| Yellow Meteor     | 34    |
| Cyan Meteor       | 35    |
| Purple Meteor     | 36    |
| White Meteor      | 37    |
| Red Wave          | 38    |
| Green Wave        | 39    |
| Blue Wave         | 40    |
| Yellow Wave       | 41    |
| Cyan Wave         | 42    |
| Purple Wave       | 43    |
| White Wave        | 44    |
| Red Green Wave    | 45    |
| Red Blue Wave     | 46    |
| Red Yellow Wave   | 47    |
| Red Cyan Wave     | 48    |
| Red Purple Wave   | 49    |
| Red White Wave    | 50    |
| Green Blue Wave   | 51    |
| Green Yellow Wave | 52    |
| Green Cyan Wave   | 53    |

| Color order              | Value |
|--------------------------|-------|
| Green Purple wave        | 54    |
| Green white wave         | 55    |
| Blue Yellow wave         | 56    |
| Blue Cyan wave           | 57    |
| Blue Purple wave         | 58    |
| Blue white wave          | 59    |
| Yellow Cyan wave         | 60    |
| Yellow Purple wave       | 61    |
| Yellow white wave        | 62    |
| Cyan Purple wave         | 63    |
| Cyan white wave          | 64    |
| Purple white wave        | 65    |
| Red Dot Pulse            | 66    |
| Green Dot Pulse          | 67    |
| Blue Dot Pulse           | 68    |
| Yellow Dot Pulse         | 69    |
| Cyan Dot Pulse           | 70    |
| Purple Dot Pulse         | 71    |
| white Dot Pulse          | 72    |
| Red Green Blank Pulse    | 73    |
| Green Blue Blank Pulse   | 74    |
| Blue Yellow Blank Pulse  | 75    |
| Yellow Cyan Blank Pulse  | 76    |
| Cyan Purple Blank Pulse  | 77    |
| Purple white Blank Pulse | 78    |
| Red with Purple Pulse    | 79    |
| Green with Cyan Pulse    | 80    |



| Color order              | Value |
|--------------------------|-------|
| Blue with Yellow Pulse   | 81    |
| Yellow with Blue Pulse   | 82    |
| Cyan with Green Pulse    | 83    |
| Purple with Purple Pulse | 84    |
| Red Comet Spin           | 85    |
| Green Comet Spin         | 86    |
| Blue Comet Spin          | 87    |
| Yellow Comet Spin        | 88    |
| Cyan Comet Spin          | 89    |
| Purple Comet Spin        | 90    |
| white Comet Spin         | 91    |
| Red Dot Spin             | 92    |
| Green Dot Spin           | 93    |
| Blue Dot Spin            | 94    |
| Yellow Dot Spin          | 95    |
| Cyan Dot Spin            | 96    |
| Purple Dot Spin          | 97    |
| white Dot Spin           | 98    |
| Red Segment Spin         | 99    |
| Green Segment Spin       | 100   |
| Blue Segment Spin        | 101   |
| Yellow Segment Spin      | 102   |
| Cyan Segment Spin        | 103   |
| Purple Segment Spin      | 104   |
| white Segment Spin       | 105   |
| Red Green Gradual Snake  | 106   |
| Red Blue Gradual Snake   | 107   |

| Color order                 | Value |
|-----------------------------|-------|
| Red Yellow Gradual Snake    | 108   |
| Red Cyan Gradual Snake      | 109   |
| Red Purple Gradual Snake    | 110   |
| Red white Gradual Snake     | 111   |
| Green Blue Gradual Snake    | 112   |
| Green Yellow Gradual Snake  | 113   |
| Green Cyan Gradual Snake    | 114   |
| Green Purple Gradual Snake  | 115   |
| Green white Gradual Snake   | 116   |
| Blue Yellow Gradual Snake   | 117   |
| Blue Cyan Gradual Snake     | 118   |
| Blue Purple Gradual Snake   | 119   |
| Blue white Gradual Snake    | 120   |
| Yellow Cyan Gradual Snake   | 121   |
| Yellow Purple Gradual Snake | 122   |
| Yellow white Gradual Snake  | 123   |
| Cyan Purple Gradual Snake   | 124   |
| Cyan white Gradual Snake    | 125   |
| Purple white Gradual Snake  | 126   |
| Red white Blank Snake       | 127   |
| Green white Blank Snake     | 128   |
| Blue white Blank Snake      | 129   |
| Yellow white Blank Snake    | 130   |
| Cyan white Blank Snake      | 131   |
| Purple white Blank Snake    | 132   |
| Green Yellow white Snake    | 133   |
| Red Green white Snake       | 134   |

| Color order                   | Value |
|-------------------------------|-------|
| Red Yellow Snake              | 135   |
| Red White Snake               | 136   |
| Green White Snake             | 137   |
| Red Stars                     | 138   |
| Green Stars                   | 139   |
| Blue Stars                    | 140   |
| Yellow Stars                  | 141   |
| Cyan Stars                    | 142   |
| Purple Stars                  | 143   |
| White Stars                   | 144   |
| Red Background Stars          | 145   |
| Green Background Stars        | 146   |
| Blue Background Stars         | 147   |
| Yellow Background Stars       | 148   |
| Cyan Background Stars         | 149   |
| Purple Background Stars       | 150   |
| Red White Background Stars    | 151   |
| Green White Background Stars  | 152   |
| Blue White Background Stars   | 153   |
| Yellow White Background Stars | 154   |
| Cyan White Background Stars   | 155   |
| Purple White Background Stars | 156   |
| White White Background Stars  | 157   |
| Red Breath                    | 158   |
| Green Breath                  | 159   |
| Blue Breath                   | 160   |
| Yellow Breath                 | 161   |

| Color order           | Value |
|-----------------------|-------|
| Cyan Breath           | 162   |
| Purple Breath         | 163   |
| White Breath          | 164   |
| Red Yellow Fire       | 165   |
| Red Purple Fire       | 166   |
| Green Yellow Fire     | 167   |
| Green Cyan Fire       | 168   |
| Blue Purple Fire      | 169   |
| Blue Cyan Fire        | 170   |
| Red Strobe            | 171   |
| Green Strobe          | 172   |
| Blue Strobe           | 173   |
| Yellow Strobe         | 174   |
| Cyan Strobe           | 175   |
| Purple Strobe         | 176   |
| White Strobe          | 177   |
| Red Blue white Strobe | 178   |
| Full Color Strobe     | 179   |

## [5] Custom effect mode

| Color order         | Value |
|---------------------|-------|
| STATIC              | 0     |
| CHASE FORWARD       | 1     |
| CHASE BACKWARD      | 2     |
| CHASE MIDDLE TO OUT | 3     |
| CHASE OUT TO MIDDLE | 4     |
| STARS               | 5     |
| BREATH              | 6     |
| COMET FORWARD       | 7     |
| COMET BACKWARD      | 8     |
| COMET MIDDLE TO OUT | 9     |
| COMET OUT TO MIDDLE | 10    |
| WAVE FORWARD        | 11    |
| WAVE BACKWARD       | 12    |
| WAVE MIDDLE TO OUT  | 13    |
| WAVE OUT TO MIDDLE  | 14    |
| STROBE              | 15    |
| SOLID FADE          | 16    |

## [6] Repitition

| Repitition | 数值 |
|------------|----|
| Today only | 0  |
| Everyday   | 1  |
| Week Days  | 2  |
| weekend    | 3  |