

# IPPOE Device Web API Interface Documentation

---

## Table of Contents

---

### [General Description](#)

1. [Login Authorization Module](#)
2. [Account Management Module](#)
3. [Dashboard/Statistics Module](#)
4. [Device Information Module](#)
5. [Device Configuration Module](#)
6. [Network Configuration Module](#)
7. [Time Configuration Module](#)
8. [Alarm Input Module](#)
9. [Alarm Output Module](#)
10. [Broadcast Configuration Module](#)
11. [Platform Integration Module](#)
12. [Firmware Upgrade Module](#)
13. [Audio Resources Module](#)
14. [LED Light Patterns Module](#)
15. [Device Testing Module](#)
16. [Historical Records Module](#)

---

## General Description

---

### Base URL

```
http://{Device IP Address}:{Port}/api/public/index.php
```

Example: `http://192.168.0.111/api/public/index.php`

### Request Headers

Except for the login interface, all interfaces require a Token in the request header:

Header Name	Type	Required	Description
Batoken	string	Yes	Token returned after successful login

### Request Header Example:

```
Batoken: eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.m/aq0H1Qcvf0BjGYrLD9Ir+w1NKCWgN...
```

## Content-Type

Regular interfaces:

```
Content-Type: application/json
```

File upload interfaces:

```
Content-Type: multipart/form-data
```

## Unified Response Format

```
{
  "code": 1,
  "msg": "",
  "data": {}
}
```

List response with pagination:

```
{
  "code": 1,
  "msg": "",
  "data": [],
  "total": 100
}
```

## Status Code Description

code	Description
1	Success
0	Failure/Error
301	Invalid Token, need to login again
403	Insufficient permissions
404	Resource not found
500	System error
501	Validation error

---

## 1. Login Authorization Module

---

## 1.1 User Login

**Request Path:** /login/login

**Request Method:** POST

**Token Required:** No

**Request Parameters:**

Parameter	Type	Required	Description
username	string	Yes	Username
password	string	Yes	Password

**Request Example:**

```
{
  "username": "admin",
  "password": "123456"
}
```

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": {
    "userInfo": {
      "id": 1,
      "username": "admin",
      "nickname": "admin",
      "avatar": "",
      "last_login_time": "",
      "token":
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.m/aq0H1Qcvf0BjGyrLD9Ir+w1NKCWgn/hfP1SD491X
+cMfIhugh0a7Dl8lBfrTNYwLdPawRutTIb1VYkDHQiu773R7sEqRBgb0XS1QvjtwhCsxEeDJrnLahTci
Oaj8g=.ZTcwMTRlNmJhNmU5NTBhODQ0N2RkODljYzRjyTlhY2U",
      "refresh_token": ""
    },
    "devInfo": {
      "ledType": "Green",
      "aiType": "CI1302-UART0",
      "aiEnable": true
    }
  }
}
```

**Error Response Example:**

```
{
  "code": 0,
  "msg": "user error",
  "data": []
}
{
  "code": 0,
  "msg": "pwd error",
  "data": []
}
```

---

## 1.2 Get Login Information

Verify Token validity and get user information and menus.

**Request Path:** `/login/info`

**Request Method:** `GET`

**Token Required:** Yes (via Header Batoken)

**Request Parameters:** None

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": {
    "adminInfo": {
      "id": 1,
      "username": "admin",
      "nickname": "admin",
      "avatar": "",
      "last_login_time": "",
      "super": true,
      "must_change_pwd": false
    },
    // Menu tree
    "menus": []
  }
}
```

**Invalid Token Response:**

```
{
  "code": 301,
  "msg": "token error",
  "data": {
    "type": "need login"
  }
}
```

---

## 1.3 User Logout

**Request Path:** `/Login/Logout`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:** None

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 1.4 Check System Time

Check device system time before login. If device time is abnormal (earlier than 2024), client time will be used for calibration.

**Request Path:** `/Login/checkTime`

**Request Method:** `POST`

**Token Required:** No

**Request Parameters:**

Parameter	Type	Required	Description
dateTime	string	No	Client current time, format: Y-m-d H:i:s
timezone	string	No	Client timezone, e.g.: UTC+8

**Request Example:**

```
{
  "dateTime": "2025-12-23 10:30:00",
  "timezone": "UTC+8"
}
```

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 2. Account Management Module

## 2.1 Change Password

Change the password of the currently logged-in user.

**Request Path:** `/account/changePwd`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
old	string	Yes	Old password
pwd	string	Yes	New password
pwd2	string	Yes	Confirm new password

**Request Example:**

```
{
  "old": "123456",
  "pwd": "newPassword123",
  "pwd2": "newPassword123"
}
```

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

**Error Response:**

```
{
  "code": 0,
  "msg": "pwd error",
  "data": []
}
{
  "code": 0,
  "msg": "Confirm Password Error",
  "data": []
}
```

---

## 2.2 Get Account List

**Request Path:** `/account/getList`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
page	number	No	Page number, default 1
limit	number	No	Items per page, default 10
quickSearch	string	No	Quick search (username prefix match)
order	string	No	Sort field
search	string	No	Search conditions

**Request Example:**

```
{
  // Search conditions
  "search": [{
    "field": "username",
    "val": "admin",
    "operator": "LIKE"
  }],
  // Quick search
  "quickSearch": "admin",
  // Current page
  "page": 1,
  // Items per page
  "limit": 10
}
```

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": [
    {
      "id": 1,
      "username": "admin",
      "user_level": "1",
      "note": ""
    },
    {
      "id": 2,
      "username": "viewer",
      "user_level": "2",

```

```
        "note": ""
      }
    ],
    "total": 2
  }
}
```

---

## 2.3 Get Account Details

**Request Path:** `/account/info`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
id	number	Yes	Account ID

**Request Example:**

```
{
  "id": 1
}
```

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": {
    "id": 1,
    "username": "admin",
    "user_level": "1",
    "note": ""
  }
}
```

---

## 2.4 Add Account

**Request Path:** `/account/add`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
username	string	Yes	Username
user_level	string	Yes	User level: 1 admin, 2 regular user

Parameter	Type	Required	Description
pwd	string	No	Password, default 123456
pwd2	string	No	Confirm password, default 123456

#### Request Example:

```
{
  "username": "operator",
  "user_level": "2",
  "pwd": "123456",
  "pwd2": "123456"
}
```

#### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

#### Error Response:

```
{
  "code": 0,
  "msg": "username already exists",
  "data": []
}
```

## 2.5 Edit Account

**Request Path:** `/account/edit`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
id	number	Yes	Account ID
user_level	string	Yes	User level

#### Request Example:

```
{
  "id": 2,
  "user_level": "1"
}
```

### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 2.6 Delete Account

**Request Path:** `/account/delete`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
ids	array	Yes	Account ID array

**Request Example:**

```
{
  "ids": [2, 3]
}
```

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": ["user1", "user2"]
}
```

Note: Returns array of deleted usernames in data field

---

## 3. Dashboard/Statistics Module

### 3.1 Get Today's Alarm Statistics

Statistics of today's alarm counts by event type.

**Request Path:** `/preview/getCount`

**Request Method:** `POST`

**Token Required:** Yes

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": {
    "gpio_in": 5,
    "ai_keywords": 3,
    "ai_sounds": 2
  }
}
```

---

### 3.3 Get Bar Chart Data

Get total alarm counts for each day in the past 7 days.

**Request Path:** `/preview/getBarData`

**Request Method:** `POST`

**Token Required:** Yes

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": {
    "xList": ["Mon", "Tue", "Wed", "Thu", "Fri", "Sat", "Sun"],
    "yList": [10, 5, 8, 12, 3, 6, 9]
  }
}
```

---

### 3.4 Get Pie Chart Data

Get proportion of alarm counts by event type for today.

**Request Path:** `/preview/getPieData`

**Request Method:** `POST`

**Token Required:** Yes

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": [
    {"name": "Gunshot", "value": 3},
    {"name": "Glass Breaking", "value": 5},
    {"name": "Scream", "value": 2},
    {"name": "Help", "value": 0}
  ]
}
```

## 3.5 Get Line Chart Data

Get daily alarm count trends by event type for the past 7 days.

**Request Path:** `/preview/getLineData`

**Request Method:** `POST`

**Token Required:** Yes

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": {
    "xList": ["Mon", "Tue", "Wed", "Thu", "Fri", "Sat", "Sun"],
    "yList": [
      {
        "name": "Gunshot",
        "type": "line",
        "data": [1, 0, 2, 1, 0, 0, 1]
      },
      {
        "name": "Glass Breaking",
        "type": "line",
        "data": [2, 1, 0, 3, 1, 2, 0]
      }
    ]
  }
}
```

---

## 4. Device Information Module

### 4.1 Get Device Information

**Request Path:** `/device/info`

**Request Method:** `POST`

**Token Required:** Yes

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": {
    "sn": "SN123456789",
    "app_version": "1.0.0",
    "fw_version": "2.1.0",
    "main_version": "1.0.0",
    "rtsp_version": "1.0.0",
    "diskFree": "1.2G",
    "mac": "AA:BB:CC:DD:EE:FF",
  }
}
```

```
    "ip": "192.168.1.100",
    "model": "",
    "name": "",
    "manager": "",
    "note": ""
  }
}
```

---

## 4.2 Save Device Information

**Request Path:** `/device/save`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
data	object	Yes	Device information object
data.name	string	No	Device name
data.manager	string	No	Administrator

**Request Example:**

```
{
  "data": {
    "name": "Hall Alarm",
    "manager": "Zhang San"
  }
}
```

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 5. Device Configuration Module

### 5.1 Get Device Configuration

**Request Path:** `/devCfg/info`

**Request Method:** `POST`

**Token Required:** Yes

## Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": {
    "cruise_led": "1",
    "cruise_led_color": "[\"blue\", \"green\"]",
    "volume_alarm": 80,
    "volume_media": 70,
    "ai_switch": "1",
    "ai_threshold_help_911": 60,
    "ai_threshold_help_me": 60,
    "ai_threshold_emergency": 60,
    "ai_threshold_scream": 60,
    "ai_threshold_gunshot": 60,
    "ai_threshold_glass": 60,
    "work_days": "[\"Mon\", \"Tue\", \"Wed\", \"Thu\", \"Fri\"]",
    "work_times": "[\"09:00\", \"18:00\"]",
    "brightness_white": 100,
    "brightness_blue": 100,
    "brightness_red_green": 100,
    "allow_other_device_event": "0",
    "alarm_release_interval": 5
  }
}
```

## 5.2 Save Device Configuration

**Request Path:** `/devCfg/save`

**Request Method:** `POST`

**Token Required:** Yes

*Explanation:*

*data.ai\_threshold\_where \* is: help\_911,help\_me,emergency,scream,gunshot,glass\**

*data.brightness\_where \* is: white,blue,red\_green\**

**Request Parameters:**

Parameter	Type	Required	Description
data	object	Yes	Configuration information object
data.cruise_led	string	No	Cruise light switch: 0 off, 1 on
data.cruise_led_color	array	No	Cruise light color array
data.volume_alarm	number	No	Alarm volume 1-10
data.volume_media	number	No	Media volume 1-10
data.ai_switch	string	No	AI switch: 0 off, 1 on

Parameter	Type	No Required	Description
data.ai_threshold_*	number	No	AI thresholds 0-100
data.work_days	array	No	Work days
data.work_times	array	No	Work time periods
data.brightness_*	number	No	Light brightness 1-10
data.alarm_release_interval	number	No	Alarm release interval (seconds)

#### Request Example:

```
{
  "data": {
    "cruise_led": "1",
    "cruise_led_color": ["blue", "green"],
    "volume_alarm": 80,
    "volume_media": 70,
    "ai_switch": "1",
    "ai_threshold_gunshot": 60
  }
}
```

#### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

## 5.3 Reboot Device

**Request Path:** `/devCfg/reboot`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:** None

#### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

## 5.4 Factory Reset

**Request Path:** `/devCfg/factory`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:** None

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

   **Warning:** After factory reset, the device will automatically reboot and all configurations will be cleared.

---

## 6. Network Configuration Module

---

### 6.1 Get Network Configuration

**Request Path:** `/network/getInfo`

**Request Method:** `POST`

**Token Required:** Yes

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": {
    "dhcp_switch": "0",
    "ipv4_addr": "192.168.1.100",
    "ipv4_mask": "255.255.255.0",
    "ipv4_gateway": "192.168.1.1",
    "mac": "AA:BB:CC:DD:EE:FF",
    "dns1": "8.8.8.8",
    "dns2": "8.8.4.4"
  }
}
```

### 6.2 Set Network Configuration

**Request Path:** `/network/setInfo`

**Request Method:** `POST`

**Token Required:** Yes

### Request Parameters:

Parameter	Type	Required	Description
dhcp_switch	string	Yes	DHCP switch: 0 off (static IP), 1 on
ipv4_addr	string	Conditionally required	IPv4 address (required when DHCP off)
ipv4_mask	string	Conditionally required	Subnet mask (required when DHCP off)
ipv4_gateway	string	Conditionally required	Gateway address (required when DHCP off)
dns1	string	No	Primary DNS
dns2	string	No	Secondary DNS

### Request Example (Static IP):

```
{
  "dhcp_switch": "0",
  "ipv4_addr": "192.168.1.100",
  "ipv4_mask": "255.255.255.0",
  "ipv4_gateway": "192.168.1.1",
  "dns1": "8.8.8.8",
  "dns2": "8.8.4.4"
}
```

### Request Example (DHCP):

```
{
  "dhcp_switch": "1"
}
```

### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

### Error Response:

```
{
  "code": 0,
  "msg": "param error",
  "data": []
}
```

---

# 7. Time Configuration Module

## 7.1 Get Time Configuration

**Request Path:** /ntp/info

**Request Method:** POST

**Token Required:** Yes

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": {
    "ntp_switch": "1",
    "ntp_address": "ntp.aliyun.com",
    "time_zone": "UTC+8",
    "dst_switch": "0",
    "system_time": "2025-12-23 10:30:00"
  }
}
```

## 7.2 Save Time Configuration

**Request Path:** /ntp/save

**Request Method:** POST

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
ntp_switch	string	Yes	NTP switch: 0 off, 1 on
ntp_address	string	Conditionally required	NTP server address (required when NTP on)
time_zone	string	Yes	Timezone, e.g.: UTC+8
dst_switch	string	No	Daylight saving time switch: 0 off, 1 on
system_time	string	No	Manually set system time (effective when NTP off)

**Request Example (Enable NTP):**

```
{
  "ntp_switch": "1",
  "ntp_address": "ntp.aliyun.com",
  "time_zone": "UTC+8",
  "dst_switch": "0"
}
```

#### Request Example (Manual time setting):

```
{
  "ntp_switch": "0",
  "time_zone": "UTC+8",
  "system_time": "2025-12-23 10:30:00"
}
```

#### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": "2025-12-23 10:30:00"
}
```

#### Error Response:

```
{
  "code": 0,
  "msg": "param error",
  "data": []
}
{
  "code": 0,
  "msg": "set ntp error",
  "data": []
}
```

---

## 8. Alarm Input Module

### 8.1 Get Alarm Input Configuration

**Request Path:** `/eventInput/info`

**Request Method:** `POST`

**Token Required:** Yes

#### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": {
```

```
"eventInfo": {
  "gpio_in": [
    {
      "id": 1,
      "event_type": "gpio_in",
      "event_name": "GPIO Input",
      "data_source": "gpio00",
      "status": "1",
      "thr_trigger": "",
      "thr_release": ""
    }
  ],
  "ai_keywords": [
    {
      "id": 2,
      "event_type": "ai_keywords",
      "event_name": "Help",
      "data_source": "help",
      "status": "1",
      "thr_trigger": "",
      "thr_release": ""
    }
  ],
  "ai_sounds": [
    {
      "id": 3,
      "event_type": "ai_sounds",
      "event_name": "Gunshot",
      "data_source": "gunshot",
      "status": "1",
      "thr_trigger": "",
      "thr_release": ""
    }
  ]
},
"otherInfo": {
  "allow_other_device_event": "0"
}
}
```

---

## 8.2 Save Alarm Input Configuration

**Request Path:** `/eventInput/save`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
eventInfo	object	Yes	Event configuration, grouped by type
otherInfo	object	Yes	Other configuration
otherInfo.allow_other_device_event	string	No	Allow other device events: 0 no, 1 yes
noiseInfo	object	No	Noise configuration
noiseInfo.status	string	No	Status
noiseInfo.thr_trigger	string	No	Trigger threshold
noiseInfo.thr_release	string	No	Release threshold

#### Request Example:

```
{
  "eventInfo": {
    "gpio_in": ["gpio00"],
    "ai_keywords": ["help", "emergency"],
    "ai_sounds": ["gunshot", "glass"]
  },
  "otherInfo": {
    "allow_other_device_event": "0"
  },
  "noiseInfo": {
    "status": "1",
    "thr_trigger": "80",
    "thr_release": "60"
  }
}
```

#### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 9. Alarm Output Module

## 9.1 Get Alarm Output Configuration

Get output configuration corresponding to enabled event inputs.

**Request Path:** `/eventOutput/info`

**Request Method:** POST

**Token Required:** Yes

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": [
    {
      "event_id": 1,
      "event_name": "Gunshot",
      "id": 1,
      "gpio_out_type": "high",
      "gpio_out_duration": "5",
      "audio_id": "/path/to/audio.wav",
      "led_color": "[\"red\", \"blue\"]",
      "led_signal_id": "1",
      "to_broadcast": "1",
      "http_enable": "1",
      "tcp_enable": "0",
      "mqtt_enable": "1"
    }
  ]
}
```

---

## 9.2 Save Single Alarm Output Configuration

**Request Path:** `/eventOutput/save`

**Request Method:** POST

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
id	number	No	Configuration ID, empty for new
event_id	number	Yes	Event ID
gpio_out_type	string	No	GPIO output type: high/low
gpio_out_duration	string	No	GPIO duration (seconds)
audio_id	string	No	Audio file path
led_color	array	No	LED color array

led_signal_id Parameter	string Type	No Required	Light pattern ID Description
to_broadcast	string	No	Broadcast: 0 no, 1 yes
http_enable	string	No	HTTP push: 0 disable, 1 enable
tcp_enable	string	No	TCP push: 0 disable, 1 enable
mqtt_enable	string	No	MQTT push: 0 disable, 1 enable

#### Request Example:

```
{
  "event_id": 1,
  "gpio_out_type": "high",
  "gpio_out_duration": "5",
  "audio_id": "/mnt/audio/alarm.wav",
  "led_color": ["red", "blue"],
  "led_signal_id": "1",
  "to_broadcast": "1",
  "http_enable": "1",
  "tcp_enable": "0",
  "mqtt_enable": "1"
}
```

#### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

## 9.3 Batch Save Alarm Output Configuration

**Request Path:** `/eventOutput/saveAll`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:** Array of configuration objects, each with same fields as 9.2

#### Request Example:

```
[
  {
    "event_id": 1,
    "gpio_out_type": "high",
    "gpio_out_duration": "5",
    "led_color": ["red"],
    "led_signal_id": "1"
  },
  {
```

```
    "event_id": 2,
    "gpio_out_type": "low",
    "gpio_out_duration": "3",
    "led_color": ["blue"],
    "led_signal_id": "2"
  }
]
```

#### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 9.4 Get Output Resource List

Get configurable output resources (LED colors, light patterns, GPIO types, audio list).

**Request Path:** `/eventOutput/getResource`

**Request Method:** `POST`

**Token Required:** Yes

#### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": {
    "ledColor": [
      {"label": "Green", "value": "green"},
      {"label": "Blue", "value": "blue"},
      {"label": "White", "value": "white"}
    ],
    "ledInfo": [
      {"value": "1", "label": "Fast Flash"},
      {"value": "2", "label": "Slow Flash"}
    ],
    "gpioOut": [
      {"label": "High", "value": "high"},
      {"label": "Low", "value": "low"}
    ],
    "audio": [
      {"label": "alarm.wav", "value": "/mnt/audio/default/alarm.wav"},
      {"label": "warning.mp3", "value": "/mnt/audio/custom/warning.mp3"}
    ]
  }
}
```

## 9.5 Test Alarm Output

Test triggering alarm output actions.

**Request Path:** `/eventOutput/test`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
type	string	Yes	Test type: trigger trigger/release release
data	object	Yes	Output configuration data
data.event_name	string	Yes	Event name
data.led_signal_id	string	No	Light pattern ID
data.http_enable	string	No	HTTP push
data.tcp_enable	string	No	TCP push
data.mqtt_enable	string	No	MQTT push

**Request Example:**

```
{
  "type": "trigger",
  "data": {
    "event_name": "Gunshot",
    "led_signal_id": "1",
    "led_color": ["red"],
    "http_enable": "1",
    "tcp_enable": "0",
    "mqtt_enable": "0"
  }
}
```

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 9.6 Restore Default Output Configuration

Delete all custom output configurations.

**Request Path:** `/eventOutput/setDefault`

**Request Method:** POST

**Token Required:** Yes

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 10. Broadcast Configuration Module

### 10.1 Get Broadcast Configuration

**Request Path:** `/broadcast/info`

**Request Method:** POST

**Token Required:** Yes

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": {
    "is_full_network": "0",
    "method": "[\"sn_list\", \"ip_list\"]",
    "sn_list": "SN001\nSN002\nSN003",
    "ip_list": "192.168.1.101\n192.168.1.102",
    "start_ip": "",
    "end_ip": ""
  }
}
```

Note: sn\_list and ip\_list separated by newline `\n`

---

### 10.2 Save Broadcast Configuration

**Request Path:** `/broadcast/save`

**Request Method:** POST

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
is_full_network	string	Yes	Full network broadcast: 0 no, 1 yes
method	array	Conditionally required	Broadcast method array (required when not full network)
sn_list	string	Conditionally required	SN list, newline separated (required when method contains sn_list)
ip_list	string	Conditionally required	IP list, newline separated (required when method contains ip_list)
start_ip	string	Conditionally required	Start IP (required when method contains ip_segment)
end_ip	string	Conditionally required	End IP (required when method contains ip_segment)

#### method options:

- `sn_list` - By SN list
- `ip_list` - By IP list
- `ip_segment` - By IP segment

#### Request Example (Full network broadcast):

```
{
  "is_full_network": "1"
}
```

#### Request Example (By SN and IP list):

```
{
  "is_full_network": "0",
  "method": ["sn_list", "ip_list"],
  "sn_list": "SN001\nSN002",
  "ip_list": "192.168.1.101\n192.168.1.102"
}
```

#### Request Example (By IP segment):

```
{
  "is_full_network": "0",
  "method": ["ip_segment"],
  "start_ip": "192.168.1.100",
  "end_ip": "192.168.1.200"
}
```

### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

### Error Response:

```
{
  "code": 0,
  "msg": "IP 192.168.1.999 format is invalid",
  "data": []
}
{
  "code": 0,
  "msg": "The number of IPs within an IP segment cannot exceed 100",
  "data": []
}
```

---

## 11. Platform Integration Module

---

### 11.1 Get Platform Configuration

**Request Path:** `/setup/info`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
type	string	Yes	Configuration type

**type options:**

- `platformHttp` - HTTP push configuration
- `platformTcp` - TCP push configuration
- `platformMqtt` - MQTT push configuration
- `heartbeat` - Heartbeat configuration

**Request Example:**

```
{
  "type": "platformHttp"
}
```

#### Response Example (HTTP configuration):

```
{
  "code": 1,
  "msg": "",
  "data": {
    "server": "http://api.example.com/alarm",
    "method": "POST",
    "content_type": "application/json",
    "retry": "3",
    "trigger_content": "{\"event\": \"${event}\", \"time\": \"${time}\"}",
    "release_content": "
    {\"event\": \"${event}\", \"time\": \"${time}\", \"type\": \"release\"}",
    "trigger_enable": "1",
    "release_enable": "1",
    "status": "1",
    "trigger_lang": "en",
    "release_lang": "en",
    "auth_method": "bearer",
    "auth_key": "Authorization",
    "auth_value": "Bearer xxx"
  }
}
```

#### Response Example (TCP configuration):

```
{
  "code": 1,
  "msg": "",
  "data": {
    "host": "192.168.1.200",
    "port": "8080",
    "retry": "3",
    "trigger_content": "ALARM:${event}",
    "release_content": "RELEASE:${event}",
    "trigger_enable": "1",
    "release_enable": "1",
    "status": "1",
    "trigger_lang": "en",
    "release_lang": "en"
  }
}
```

#### Response Example (MQTT configuration):

```
{
  "code": 1,
  "msg": "",
  "data": {
```

```

    "server": "mqtt.example.com",
    "port": "1883",
    "client_id": "device_001",
    "user": "admin",
    "password": "*****",
    "topic": "alarm/events",
    "qos": "1",
    "trigger_content": "{\"event\": \"${event}\"}",
    "release_content": "{\"event\": \"${event}\", \"type\": \"release\"}",
    "trigger_enable": "1",
    "release_enable": "1",
    "status": "1",
    "trigger_lang": "en",
    "release_lang": "en"
  }
}

```

### Response Example (Heartbeat configuration):

```

{
  "code": 1,
  "msg": "",
  "data": {
    "content": "{\"sn\": \"${sn}\", \"status\": \"online\"}",
    "protocol": "http",
    "status": "1",
    "interval": "60",
    "lang": "en",
    "mqtt_conf": "{...}",
    "tcp_conf": "{...}",
    "http_conf": "{...}"
  }
}

```

## 11.2 Save Platform Configuration

**Request Path:** `/setup/save`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
type	string	Yes	Configuration type (same as 11.1)
data	object	Yes	Configuration data (fields same as 11.1 response)

**Request Example (HTTP configuration):**

```

{
  "type": "platformHttp",

```

```
"data": {
  "server": "http://api.example.com/alarm",
  "method": "POST",
  "content_type": "application/json",
  "retry": "3",
  "trigger_content": "{\"event\": \"${event}\"}",
  "release_content": "{\"event\": \"${event}\", \"type\": \"release\"}",
  "trigger_enable": "1",
  "release_enable": "1",
  "status": "1"
}
```

#### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 11.3 HTTP Connection Test

**Request Path:** /platformTest/httpTest

**Request Method:** POST

**Token Required:** Yes

#### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 11.4 TCP Connection Test

**Request Path:** /platformTest/tcpTest

**Request Method:** POST

**Token Required:** Yes

#### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 11.5 MQTT Connection Test

**Request Path:** `/platformTest/mqttTest`

**Request Method:** `POST`

**Token Required:** Yes

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 11.6 Heartbeat Test

**Request Path:** `/platformTest/heartTest`

**Request Method:** `POST`

**Token Required:** Yes

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

# 12. Firmware Upgrade Module

---

## 12.1 Get Upgrade Status

Check if device can currently upload/upgrade firmware.

**Request Path:** `/upgrade/getStatus`

**Request Method:** `POST`

**Token Required:** Yes

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 12.2 Upload Firmware

**Request Path:** `/upgrade/upload`

**Request Method:** `POST`

**Content-Type:** `multipart/form-data`

**Token Required:** Yes

**Recommended Timeout:** 60 seconds

**Request Parameters:**

Parameter	Type	Required	Description
file	File	Yes	Firmware file (.bin format, max 20MB)

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

**Error Response:**

```
{
  "code": 0,
  "msg": "Please upload the specified type",
  "data": []
}
{
  "code": 0,
  "msg": "The file size cannot exceed 10MB",
  "data": []
}
```

---

## 12.3 Get Firmware List

**Request Path:** `/upgrade/getList`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
page	number	No	Page number
size	number	No	Items per page, default 15

### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": [
    {
      "id": 1,
      "version": "2.1.0",
      "md5": "abc123def456...",
      "username": "admin",
      "status": "-1",
      "ctime": 1703318400,
      "uptime": "2025-12-23 10:00:00",
      "note": "firmware_v2.1.0.bin"
    }
  ],
  "total": 1
}
```

### status description:

Value	Description
-1	Not upgraded
0	Upgrade successful
1	Upgrading
9	Upgrade failed

## 12.4 Execute Upgrade

**Request Path:** `/upgrade/upgrade`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
md5	string	Yes	Firmware MD5 value

**Request Example:**

```
{
  "md5": "abc123def456..."
}
```

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 12.5 Get Upgrade Progress

**Request Path:** `/upgrade/getStep`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
md5	string	Yes	Firmware MD5 value

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": {
    "status": "1",
    "progress": 50
  }
}
```

---

## 13. Audio Resources Module

### 13.1 Upload Audio File

**Request Path:** `/audio/upload`

**Request Method:** `POST`

**Content-Type:** `multipart/form-data`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
file	File	Yes	Audio file (.wav/.mp3 format, max 10MB)

**Filename requirements:** Only numbers, letters and underscores allowed

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

#### Error Response:

```
{
  "code": 0,
  "msg": "File names only allow numbers, letters, and underscores",
  "data": []
}
```

---

## 13.2 Get Audio List

**Request Path:** `/audio/getList`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
type	string	Yes	Type: default system default/custom custom
page	number	No	Page number, default 1
size	number	No	Items per page, default 10

#### Request Example:

```
{
  "type": "custom",
  "page": 1,
  "size": 10
}
```

#### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": [
    {
      "filename": "alarm_sound.wav",
      "uptime": "2025-12-23 10:00:00"
    }
  ],
  "total": 1
}
```

---

## 13.3 Delete Audio

**Request Path:** `/audio/delete`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
ids	array	Yes	Filename array

**Request Example:**

```
{
  "ids": ["alarm_sound.wav", "warning.mp3"]
}
```

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 13.4 Play Audio Test

**Request Path:** `/audio/play`

**Request Method:** `POST`

**Token Required:** Yes

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 13.5 Stop Audio Playback

**Request Path:** `/audio/stop`

**Request Method:** `POST`

**Token Required:** Yes

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 14. LED Light Patterns Module

### 14.1 Get Light Pattern List

**Request Path:** `/led/getList`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
tab	string	No	Tab: system system/custom custom
page	number	No	Page number, default 1
limit	number	No	Items per page, default 10
quickSearch	string	No	Name quick search
order	string	No	Sort order

**Request Example:**

```
{
  "tab": "custom",
  "page": 1,
  "limit": 10
}
```

### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": [
    {
      "id": 1,
      "type": "flash",
      "name": "Fast Flash",
      "total_duration": 10,
      "each_duration": "100",
      "direction": "forward",
      "status": "1",
      "is_system": "0"
    }
  ],
  "total": 1
}
```

Note: total\_duration returns in seconds (stored as milliseconds in database)

## 14.2 Get Enabled Light Pattern Options

For dropdown selection.

**Request Path:** `/led/getOptions`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
flag	string	No	Filter: all/all/system/system/custom/custom, default all

### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": [
    {"id": "1", "name": "Fast Flash"},
    {"id": "2", "name": "Slow Flash"},
    {"id": "3", "name": "Breathing Light"}
  ]
}
```

## 14.3 Get Light Pattern Details

**Request Path:** `/led/info`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
id	number	Yes	Light pattern ID

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": {
    "id": 1,
    "type": "flash",
    "name": "Fast Flash",
    "total_duration": 10,
    "each_duration": "100",
    "direction": "forward",
    "status": "1",
    "is_system": "0"
  }
}
```

## 14.4 Set Light Pattern Status

Batch enable/disable light patterns.

**Request Path:** `/led/setStatus`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
ids	array	Yes	Light pattern ID array
status	string	Yes	Status: 0 disable, 1 enable
type	string	Yes	Type identifier

**Request Example:**

```
{
  "ids": [1, 2, 3],
  "status": "1",
  "type": "custom"
}
```

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 14.5 Add Light Pattern

**Request Path:** `/led/add`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
name	string	Yes	Light pattern name
type	string	Yes	Light pattern type
total_duration	number	No	Total duration (seconds)
each_duration	string	No	Each duration (milliseconds)
direction	string	No	Direction

**Request Example:**

```
{
  "name": "Custom Flash",
  "type": "flash",
  "total_duration": 10,
  "each_duration": "200",
  "direction": "forward"
}
```

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

**Error Response:**

```
{
  "code": 0,
  "msg": "name already exists",
  "data": []
}
```

---

## 14.6 Edit Light Pattern

**Request Path:** `/led/edit`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
id	number	Yes	Light pattern ID
name	string	Yes	Light pattern name
type	string	Yes	Light pattern type
total_duration	number	No	Total duration (seconds)
each_duration	string	No	Each duration (milliseconds)
direction	string	No	Direction

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 14.7 Delete Light Pattern

**Request Path:** `/led/delete`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
ids	array	Yes	Light pattern ID array

**Request Example:**

```
{
  "ids": [2, 3]
}
```

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 14.8 Trigger LED Light Pattern

**Request Path:** `/led/set`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:** Light pattern configuration data

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 15. Device Testing Module

## 15.1 Play Alarm Audio

**Request Path:** `/test/playAudio`

**Request Method:** `POST`

**Token Required:** Yes

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 15.2 Stop Audio Playback

**Request Path:** `/test/stopAudio`

**Request Method:** `POST`

**Token Required:** Yes

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 15.3 LED Test

**Request Path:** `/test/setLed`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
value	string	No	Brightness value, default 5

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 15.4 Fan Test (Hardware Not Supported in This Version)

**Request Path:** `/test/setFan`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:** Fan configuration

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 15.5 GPIO High Level Test (Hardware Not Supported in This Version)

**Request Path:** `/audio/upGpio`

**Request Method:** `POST`

**Token Required:** Yes

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 15.6 GPIO Low Level Test (Hardware Not Supported in This Version)

**Request Path:** `/audio/downGpio`

**Request Method:** `POST`

**Token Required:** Yes

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": []
}
```

---

## 16. Historical Records Module

### 16.1 Get Event Logs

**Request Path:** `/eventLog/getList`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
page	number	No	Page number
limit	number	No	Items per page
startDate	string	No	Start date
endDate	string	No	End date
type	string	No	Event type

**Response Example:**

```
{
  "code": 1,
  "msg": "",
  "data": [
    {
      "id": 1,
      "event_name": "Gunshot",
      "event_type": "ai_sounds",
      "type": "trigger",
      "date": "2025-12-23",
      "ctime": 1703318400
    }
  ],
  "total": 100
}
```

### 16.2 Get Data List

**Request Path:** `/dataList/getList`

**Request Method:** `POST`

**Token Required:** Yes

**Request Parameters:**

Parameter	Type	Required	Description
page	number	No	Page number

Parameter	Type	Required	Description
-----------	------	----------	-------------

### Response Example:

```
{
  "code": 1,
  "msg": "",
  "data": [],
  "total": 0
}
```

---

## Additional Notes

### File Upload Best Practices

1.

#### Audio Files:

- Supported formats: .wav, .mp3
- Maximum size: 10MB
- Filename restrictions: Only numbers, letters, and underscores

2.

#### Firmware Files:

- Supported format: .bin
- Maximum size: 20MB
- Recommended timeout: 60 seconds

## Error Handling

### Common Error Codes:

- **0**: General failure/error
- **301**: Token invalid - redirect to login
- **403**: Insufficient permissions
-

- `404` : Resource not found

- `500` : Internal server error

- `501` : Validation error

### Error Response Structure:

```
{
  "code": 0,
  "msg": "Error description",
  "data": []
}
```

## Authentication Flow

1.

**Login:** POST `/login/login` with credentials

2.

**Store Token:** Save token from response for subsequent requests

3.

**Header:** Include token in `Batoken` header for all authenticated requests

4.

**Token Validation:** Use `/login/info` to verify token validity

5.

**Logout:** POST `/login/logout` to invalidate token

## Timezone Handling

- 

All datetime parameters should use format: `Y-m-d H:i:s`

- 

Timezone should be specified as `UTC+X` or `UTC-X`

- 

Device time can be calibrated via `/login/checkTime`

## Pagination Parameters

For list endpoints supporting pagination:

- 

`page`: Page number (starting from 1)

- 

`limit` / `size`: Number of items per page

-

Response includes `total` field with total count

## Configuration Tips

1.

**Device Configuration:** Changes may require device reboot

2.

**Network Configuration:** IP changes may disconnect current session

3.

**Factory Reset:** Irreversible - all configurations will be lost

4.

**Firmware Upgrade:** Ensure stable power during upgrade process

---

**Document Version:** 1.0

**Last Updated:** 2025-12-23

**Compatible Firmware Version:** 1.0.0+

This documentation provides comprehensive coverage of the IPPOE device Web API interface. For additional support or clarification on specific endpoints, please refer to the device manufacturer's technical support resources.