

Modator 2U

前言

| | |
|-------|-----|
| 文档简介 | 1.1 |
| 接口约定 | 1.2 |
| 测试工具 | 1.3 |
| 公共错误码 | 1.4 |
| 名词解释 | 1.5 |

机箱信息

| | |
|-----------|------|
| 获取风扇控制板信息 | 2.1 |
| 获取电源板信息 | 2.2 |
| 获取警报信息 | 2.3 |
| 获取模块信息 | 2.4 |
| 获取蜂鸣器状态 | 2.5 |
| 设置蜂鸣器 | 2.6 |
| 获取电源模块状态 | 2.7 |
| 设置电源模块 | 2.8 |
| 获取触摸屏休眠状态 | 2.9 |
| 设置触摸屏休眠状态 | 2.10 |

系统通用设置

| | |
|----------|-----|
| 定时自动重启设备 | 3.1 |
| 获取设备信息 | 3.2 |
| 获取设备基本信息 | 3.3 |
| 设置日期和时间 | 3.4 |
| 设置设备名称 | 3.5 |
| 设置时区 | 3.6 |

网络管理

| | |
|-----------|-----|
| 获取DNS | 4.1 |
| 获取网卡信息 | 4.2 |
| 配置以太网 | 4.3 |
| 设置DNS | 4.4 |
| 配置USB NET | 4.5 |

固件升级

| | |
|-------------|-----|
| 清除升级状态 | 5.1 |
| 获取固件版本和升级状态 | 5.2 |
| 执行固件升级操作 | 5.3 |
| 上传固件 | 5.4 |

用户管理

| | |
|--------|-----|
| 添加用户 | 6.1 |
| 修改登录密码 | 6.2 |

| | |
|----------|------|
| 删除用户 | 6.3 |
| 获取系统用户列表 | 6.4 |
| 用户登录 | 6.5 |
| 退出登录 | 6.6 |
| 重置用户密码 | 6.7 |
| 获取Token | 6.8 |
| 添加Token | 6.9 |
| 删除Token | 6.10 |

日志管理

| | |
|------|-----|
| 清除日志 | 7.1 |
| 导出日志 | 7.2 |
| 筛选日志 | 7.3 |

通用接口

| | |
|---------|-----|
| 重置设备鉴权 | 8.1 |
| Ping 测试 | 8.2 |
| 重启设备 | 8.3 |
| 重置设备 | 8.4 |

证书

| | |
|-----------|-----|
| 获取证书信息 | 9.1 |
| 启用HTTPS访问 | 9.2 |
| 导入SSL证书 | 9.3 |
| 删除SSL证书 | 9.4 |

文档简介

针对Modator 2U，我们开放了丰富的 API，方便开发人员与设备交互，如获取设备的基本信息（设备名称、固件版本等），修改设备配置，更新固件等。这些 API 基于 HTTP 协议，是一种轻量级、无连接状态的接口，响应数据为 JSON 格式。通过本文档，您可以更详细地了解每个 API 的功能和请求方式。

本文档中的 API 可适用于以下产品：

- Modator 2U

接口约定

概述

本篇介绍了 API 的请求数据和返回数据的构成元素，通过这些元素你可以了解如何发起请求，如何解读返回数据。

- **请求协议:** [HTTP](#)
- **请求方法:** 默认情况下，数据请求、提交及文件上传都用 [POST](#) 方式，个别 API 同时支持 [GET](#) 方式，如：[重置设备鉴权](#)，具体由各功能接口决定。
- **请求域名:** 设备网卡 IP (IPv4) 地址，如：`http://192.168.66.1`
- **请求头 (Request Headers) :** 以请求头 ([HTTP Header](#)) 提供有关请求的额外信息，以下是本文档中 API 的请求中使用的实例
 - `Content-Type` : 内容类型，用于定义网络文件的类型和网页的编码。

API 中主要包括以下两类语法格式:

1. `Content-Type: application/json; charset=utf-8` , 用于数据请求或提交, 使用 UTF-8 编码并以 [JSON](#) 格式封装。
 2. `Content-Type: multipart/form-data; boundary=${boundary}` , 用于文件上传, 支持图片、视频、音乐等媒体文件及产品固件。
- `Cookie` : 储存在用户本地终端上的数据, 在 API 中被用于辨别用户身份, 如 `Cookie: sid-A506220808450=6062n8wqjz9mm7m7op91a05j58bup424`
 - **请求参数 (Request Body) :** 允许以消息体 ([HTTP Body](#)) 的形式将附加数据传递给 API。这些参数可以是必要的, 也可以是可选的, 具体由各功能接口决定。
 - **返回数据 (Response Body) :** 以消息体 ([HTTP Body](#)) 的形式返回数据。HTTP 状态为 200 时, 返回 JSON 格式的数据, 否则为 HTTP 对应错误码。
 - **登录认证方式:** 在 `Cookie` 中携带 `sid-xx=xxxxxxxxxx` 。个别 API 无需认证, 如: [Ping测试](#)。
 - **[xxx]:** 某类数据的替代标识。
 - [IP], 设备网卡IP地址, 使用时需替换实际内容, 如: 192.168.66.1
 - [port], 端口号, 使用时需替换为实际内容, 如: 8080。
 - [serial number], 设备序列号, 使用时需替换为实际内容, 如: A506220808450。

提交请求

数据请求及提交

数据请求及提交时, 要求附加数据使用 UTF-8 编码并以 JSON 格式封装, 如:

示例 1: 添加用户 (需要登录认证)

Request Headers

```
POST / HTTP/1.1
Host: 192.168.66.1/api/user/add
Content-Type: application/json; charset=utf-8
Cookie: sid-A506220808450=6062n8wqjz9mm7m7op91a05j58bup424
```

Request Body

```
{
  "username": "test",
  "password": "c4ca4238a0b923820dcc509a6f75849b"
}
```

示例2: 测试设备网络连接 (无需登录认证)

Request Headers

```
POST / HTTP/1.1
Host: 192.168.66.1/api/ping
Content-Type: application/json; charset=utf-8
```

Request Body

None

文件上传

文件上传时，要求附件数据以二进制形式封装，如：

Request Headers

```
POST / HTTP/1.1
Host: 192.168.66.1/mwapi/upload-source-file
Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryuCSyGCncblUp3ed
Cookie: sid-A506220808450=6062n8wqjz9mm7m7op91a05j58bup424
```

Request Body

```
-----WebKitFormBoundaryuCSyGCncblUp3ed
Content-Disposition: form-data; name="file"; filename="204.jpeg"
Content-Type: application/octet-stream
XXXX
XXXX
-----WebKitFormBoundaryuCSyGCncblUp3ed--
```

返回数据

返回数据均为 JSON 数据格式。

JSON 对象中的 `status / result` 属性为 [公共错误码](#)，为 `0` 时表示数据获取或操作成功，否则为相应的失败错误码。

示例1：单层的数据封装

```
{
  "status": 0,
  "code": "Success",
  "enable": true,
  "enable-web-control": true
}
```

示例2：带有二级对象的数据封装

```
{
  "result": 0,
  "info": {}
}
```

示例3：二级对象为数组的数据封装

```
{
  "status": 0,
  "items": []
}
```

示例4：出现业务错误的数据封装

```
{
  "status": 16,
  "message": "Item is not exist."
}
```

登录认证及 API 访问身份验证

为保障系统安全，本文档中的 API 大多数需要进行身份验证，部分 API 要求管理员才可访问，如：[添加用户](#)、[删除用户](#)等。

登录认证

使用用户名 `username` 和密码 `password` 登录。

登录成功后会在 Cookie 存放 Session ID:

```
Cookie: sid-[serial number]=6440wa6u5fw8wv43f91v55cqkctnpv6
```

Session ID 持续有效，直到设备关机或重启。

Request Headers

```
POST / HTTP/1.1
Host: 192.168.66.1//api/user/login
Content-Type: application/json; charset=utf-8
```

Request Body

```
{
  "username": "test",
  "password": "c4ca4238a0b923820dcc509a6f75849b"
}
```

Response Headers

```
Content-Type: application/json; charset=utf-8
Expires: 0
Set-Cookie: sid-A506220808450=6062n8wqjz9mm7m7op91a05j58bup424
```

Response Body

```
{
  "status": "test",
  "sid": "6440wa6u5fw8wv43f91v55cqkctnpv6"
}
```

API 访问身份验证

需要身份验证的 API 访问时，需要在 Cookie 中传输登录认证接口获取的 Session ID

Request Headers

```
POST / HTTP/1.1
Host: 192.168.66.1//api/user/add
Content-Type: application/json; charset=utf-8
Cookie: sid-A506220808450=6062n8wqjz9mm7m7op91a05j58bup424
```

测试工具

在不同操作系统中，可以安装 `wget` 和 `curl` 两个工具，安装后可以在命令行中通过 `wget` 或 `curl` 命令来调用 API。

不同操作系统中，下边示例的 `cookie` 文件存放位置不同，请根据实际情况修改。(以下示例基于Linux。)

wget

1.登录并保存 cookies

```
wget --save-cookies=sid.txt --keep-session-cookies --header="Content-Type: application/json" --post-data='{"username":"Admin", "password": "c1c224b03cd9bc7b6a86d77f5dace40191766c485cd55dc48caf9ac873335d6f"}' http://192.168.66.1/api/user/login -d -q -O -
```

2.获取用户列表

```
wget --load-cookies=sid.txt --keep-session-cookies --header="Content-Type: application/json" --post-data='' http://192.168.66.1/api/user/get-all -d -q -O -
```

3.新增用户

```
wget --load-cookies=sid.txt --keep-session-cookies --header="Content-Type: application/json" --post-data='{"username":"test","password":"9f86d081884c7d659a2feaa0c55ad015a3bf4f1b2b0b822cd15d6c15b0f00a08"}' http://192.168.66.1/api/user/add -d -q -O -
```

curl

1.登录并保存 cookies

```
curl --cookie-jar sid.txt http://192.168.66.1/api/user/login -X POST -H 'Content-Type: application/json' -d '{"username":"Admin", "password": "c1c224b03cd9bc7b6a86d77f5dace40191766c485cd55dc48caf9ac873335d6f"}'
```

2.获取用户列表

```
curl --cookie sid.txt http://192.168.66.1/api/user/get-all -X POST -H 'Content-Type: application/json' -d ''
```

3.新增用户

```
curl --cookie sid.txt http://192.168.66.1/api/user/add -X POST -H 'Content-Type: application/json' -d '{"username":"test","password":"9f86d081884c7d659a2feaa0c55ad015a3bf4f1b2b0b822cd15d6c15b0f00a08"}'
```

公共错误码

| 错误码 | 定义 | 状态描述 | 说明 |
|-----|-----------------------------|-----------------|---------|
| 0 | MW_STATUS_SUCCESS | Success | 成功 |
| 1 | MW_STATUS_PENDING | Pending | 操作挂起 |
| 2 | MW_STATUS_TIMEOUT | Timeout | 操作超时 |
| 3 | MW_STATUS_INTERRUPTED | Interrupted | 操作中断 |
| 4 | MW_STATUS_TRY_AGAIN | TryAgain | 操作重试 |
| 5 | MW_STATUS_NOT_IMPLEMENTED | NotImplemented | 未实施 |
| 6 | MW_STATUS_UNKNOWN_ERROR | UnknownError | 未知错误 |
| 7 | MW_STATUS_INVALID_ARG | InvalidArgument | 参数错误 |
| 8 | MW_STATUS_NO_MEMORY | OutOfmemory | 内存不足 |
| 9 | MW_STATUS_UNSUPPORTED | Unsupported | 未提供服务 |
| 10 | MW_STATUS_FILE_BUSY | FileBusy | 文件系统繁忙 |
| 11 | MW_STATUS_DEVICE_BUSY | DeviceBusy | 设备繁忙 |
| 12 | MW_STATUS_DEVICE_LOST | DeviceLost | 设备丢失 |
| 13 | MW_STATUS_IO_FAILED | IOError | IO 错误 |
| 14 | MW_STATUS_READ_FAILED | ReadError | IO 读取失败 |
| 15 | MW_STATUS_WRITE_FAILED | WriteError | IO 写入失败 |
| 16 | MW_STATUS_NOT_EXIST | NotExist | 内容不存在 |
| 17 | MW_STATUS_TOO_MANY | TooMany | 个数超限 |
| 18 | MW_STATUS_TOO_LARGE | TooLarge | 大小超限 |
| 19 | MW_STATUS_OVERFLOW | Overflow | 溢出 (上) |
| 20 | MW_STATUS_UNDERFLOW | Underflow | 溢出 (下) |
| 21 | MW_STATUS_FORMAT_ERROR | FormatError | 格式错误 |
| 22 | MW_STATUS_FILE_EXISTS | FileExists | 文件已存在 |
| 23 | MW_STATUS_FILE_TYPE_ERROR | FileTypeError | 文件类型错误 |
| 24 | MW_STATUS_DEVICE_TYPE_ERROR | DeviceTypeError | 设备类型错误 |
| 25 | MW_STATUS_IS_DIRECTORY | IsDirectory | 目录已存在 |
| 26 | MW_STATUS_READ_ONLY | ReadOnly | 只读限制 |
| 27 | MW_STATUS_RANGE_ERROR | OutOfRange | 区间错误 |
| 28 | MW_STATUS_BROKEN_PIPE | BrokenPipe | PIPE异常 |
| 29 | MW_STATUS_NO_SPACE | NoSpace | 空间不足 |
| 30 | MW_STATUS_NOT_DIRECTORY | NotDirectory | 目录不存在 |
| 31 | MW_STATUS_NOT_PERMITTED | NotPermitted | 不允许的操作 |
| 32 | MW_STATUS_BAD_ADDRESS | BadAddress | 地址错误 |
| 33 | MW_STATUS_SEEK_ERROR | SeekError | 跳转错误 |
| 34 | MW_STATUS_CROSS_DEVICE_LINK | CrossDeviceLink | 跨设备链路错误 |
| 35 | MW_STATUS_NOT_INITIALIZED | NotInitialized | 未初始化 |

| 错误码 | 定义 | 状态描述 | 说明 |
|-----|--------------------------------|--------------------|-------|
| 36 | MW_STATUS_AUTH_FAILED | AuthFailed | 鉴权失败 |
| 37 | MW_STATUS_NOT_LOGGED_IN | NotLoggedIn | 未登录 |
| 38 | MW_STATUS_WRONG_STATE | WrongState | 状态错误 |
| 39 | MW_STATUS_MISMATCH | Mismatch | 不匹配 |
| 40 | MW_STATUS_VERIFY_FAILED | VerifyFailed | 验证失败 |
| 41 | MW_STATUS_CONSTRAINT_VIOLATION | ConstraintViolatin | 违反约束 |
| 42 | MW_STATUS_CANCELED | Canceled | 已取消 |
| 43 | MW_STATUS_IN_PROGRESS | InProgress | 进行中 |
| 44 | MW_STATUS_CONN_REFUSED | ConnectionRefused | 连接拒绝 |
| 45 | MW_STATUS_CONN_RESET | ConnectionReset | 连接重置 |
| 46 | MW_STATUS_ADDR_IN_USE | AddressInUse | 地址已占用 |
| 47 | MW_STATUS_NO_RESPONSE | NoResponse | 无响应 |
| 48 | MW_STATUS_INFO_CHANGED | InfoChanged | 内容变更 |
| 49 | MW_STATUS_INVALID_DATA | InvalidData | 非法数据 |
| 50 | MW_STATUS_NEED_MORE_DATA | NeedMoreData | 数据缺失 |
| 51 | MW_STATUS_NO_BUFFER | NoBuffer | 缓存耗尽 |
| 52 | MW_STATUS_BUFFER_TOO_SMALL | BufferTooSmall | 缓存过小 |
| 53 | MW_STATUS_BUFFER_IS_EMPTY | BufferIsEmpty | 缓存为空 |
| 54 | MW_STATUS_BUFFER_IS_FULL | BufferIsFull | 缓存已满 |

名词解释

HTTP: 超文本传输协议 (Hypertext Transfer Protocol) 是一种用于分布式、协作式和超媒体信息系统的应⤵用层协议, 是万维网 WWW (World Wide Web) 的数据通信的基础。

GET: HTTP 请求方式的一种, 常用于从服务器获取数据。其请求的数据量较小, 参数通过 URL 传递。

POST: HTTP 请求方式的一种, 主要用于向服务器提交数据。POST 方法将数据放置在请求体中, 可请求的数据量较大, 安全性更高。

HTTP Header: HTTP 请求和响应中传递元数据的一部分, 它们是一些键值对, 其中包含了关于客户端、服务器、请求和响应的各种元数据。常见的 HTTP 头有:

- Content-Type: 指定消息体的媒体类型。
- Content-Length: 指定消息体的字节长度。
- Cookie: 向服务器传输 cookie。
- Set-Cookie: 向客户端设置 cookie。

HTTP Body: HTTP 请求或响应中的消息体部分, 正文类型可以是任何类型的数据, 如文本、JSON、XML、HTML、图片、视频等。本系统多使用 JSON。

JSON: JavaScript Object Notation, 是一种基于文本的轻量级数据交换格式。JSON 独立于任何语言, 以易于人类阅读和编写的文本格式存储和表示数据, 同时也便于机器解析和生成, 能有效地提升网络传输效率。JSON 格式的数据具有良好的通用性和移植性, 在数据的表达上采用“名称/值对”的方式, 同时支持数组的表示。

sha256: sha256 是一种常用的哈希算法, 用于生成数字指纹, 广泛用于密码学和数字完整性校验等领域。

获取风扇控制板信息

1.接口描述

接口说明：获取风扇控制板信息。

请求方式：POST [ip]/api/chassis/get-fan

| 管理员权限 | 登录认证 |
|-------|------|
| 否 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|-------------|-------------------------------------|--|
| status | Int | 状态码 |
| code | String | 状态描述 |
| device-name | String | 设备名称 |
| fan-num | Int | 风扇控制板的风扇个数 |
| sensor-num | Int | 风扇控制板的温度传感器个数 |
| state | Int | 风扇控制板的状态 0: 未通电 1: 通信失败 2: 通信成功 3: 固件升级中 |
| fan | Array of FanData | 风扇信息 |
| sensor | Array of SensorData | 传感器信息 |

FanData

| 参数名称 | 类型 | 描述 |
|--------|-----|-------------------|
| id | Int | 风扇id |
| speed | Int | 风扇转速,单位rpm (转每分钟) |
| status | Int | 状态码 |

SensorData

| 参数名称 | 类型 | 描述 |
|-------------|--------|-------------------|
| id | Int | 传感器id |
| temperature | Double | 温度传感器读取的温度, 单位摄氏度 |
| status | Int | 状态码 |

4.示例

获取风扇控制板信息。

输入示例

```
None
```

输出示例

```
{
  "device-name": "Modator 2U FAN CONTROL",
  "fan-num": 5,
  "sensor-num": 3,
  "state": 2,
  "fan": [
    {
      "id": 1,
      "speed": 2250,
      "status": 0
    },
    {
      "id": 2,
      "speed": 0,
      "status": 6
    },
    {
      "id": 3,
      "speed": 0,
      "status": 6
    },
    {
      "id": 4,
      "speed": 2430,
      "status": 0
    },
    {
      "id": 5,
      "speed": 2100,
      "status": 0
    }
  ],
  "sensor": [
    {
      "id": 1,
      "temperature": 28.125,
      "status": 0
    },
    {
      "id": 2,
      "temperature": 27.0,
      "status": 0
    },
    {
      "id": 3,
      "temperature": 27.25,
      "status": 0
    }
  ],
  "status": 0,
  "code": "Success"
}

{
  "device-name": "Modator 2U FAN CONTROL",
  "fan-num": 5,
  "sensor-num": 3,
  "state": 0,
  "status": 0,
}
```

```
"code": "Success"
}
```

5. 错误码

无接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|---------------------|--|
| 4 | MW_STATUS_TRY_AGAIN | Array of FanData ，风扇尝试再次启动 |

获取电源板信息

1.接口描述

接口说明：获取与机箱连接的电源板的信息。

请求方式：POST [ip]/api/chassis/get-backplane

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 否 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|-------------|------------------------------------|--|
| status | Int | 状态码 |
| code | String | 状态描述 |
| device-name | String | 设备名称 |
| power-num | Int | 电源模块的数量 |
| state | Int | 电源板当前的状态 0: 通信失败 1: 通信成功 2: 固件升级中 |
| power | Array of PowerData | 电源模块信息 |

PowerData

| 参数名称 | 类型 | 描述 |
|-------|--------|--|
| id | Int | 电源模块的id |
| state | String | 电源模块的状态，包括ON、OFF、ERROR、No AC power和Power module disconnected五种状态 |

4.示例

获取电源板信息。

输入示例

None

输出示例

```
{
  "device-name": "Modator 2U POWER BOARD",
  "power-num": 2,
  "state": 2,
  "power": [
    {
      "id": 1,
      "state": "OFF"
    },
    {
      "id": 2,
      "state": "OFF"
    }
  ],
  "status": 0,
  "code": "Success"
}

{
  "device-name": "Modator 2U POWER BOARD",
  "power-num": 2,
  "state": 0,
  "status": 0,
  "code": "Success"
}
```

5. 错误码

无接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

获取警报信息

1.接口描述

接口说明：获取机箱的警报信息。

请求方式：POST [ip]/api/chassis/get-alert

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 否 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|-----------|------------------------------------|----------|
| status | Int | 状态码 |
| code | String | 状态描述 |
| alert-num | Int | 警报信息的总数量 |
| alert | Array of AlertData | 警报信息列表 |

AlertData

| 参数名称 | 类型 | 描述 |
|-------------|--------|---------|
| index | Int | 警报信息的索引 |
| information | String | 警报信息 |

4.示例

获取设备警报信息。

输入示例

```
None
```


输出示例

```
{
  "alert-num": 3,
  "alert": [
    {
      "index": 1,
      "information": "2023/05/29 19:38:18\npower1 DC output below minimum detection level"
    },
    {
      "index": 2,
      "information": "2023/05/29 19:38:18\nfan 2 failed to start and stopped rotating"
    },
    {
      "index": 3,
      "information": "2023/05/29 19:38:18\nfan 3 failed to start and stopped rotating"
    }
  ],
  "status": 0,
  "code": "Success"
}
```

5. 错误码

无接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

获取模块信息

1.接口描述

接口说明：获取模块信息。

请求方式：POST [ip]/api/chassis/get-card

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 否 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|----------|-----------------------------------|--------|
| status | Int | 状态码 |
| code | String | 状态描述 |
| card-num | Int | 设备槽位数 |
| card | Array of CardData | 模块信息列表 |

CardData

| 参数名称 | 类型 | 描述 |
|----------|-----------------|---|
| position | Int | 模块所在槽位 |
| connect | Boolean | 模块是否连接 true：是；false：否 |
| device | Array of device | 详情请参考 解码板设备信息 编码板设备信息 |
| ndi | Array of ndi | 详情请参考 解码板NDI状态信息 编码板NDI状态信息 |

4.示例

获取板卡模块信息。

输入示例

None

输出示例

```
{
  "card-num": 10,
  "card": [
    {
      "position": 1,
      "connect": true,
      "device": {
        "device-name": "NJ-play06",
        "model": "NDI to HDMI",
        "product-id": 1040,
        "serial-no": "Z410221102004",
        "hw-revision": "Z",
        "fw-version": "1.1.918",
        "board-id": 1,
        "cpu-usage": 2.5600000000000001,
        "memory-usage": 12.869999999999999,
        "core-temp": 36.0,
        "up-time": 25926,
        "output-state": "unconnected",
        "ip-addr": "169.254.5.48"
      },
      "ndi": {
        "name": "SRT Stream",
        "url": "srt://47.122.24.9:8000?streamid=play.srt/live/31&mode=caller&latency=125&mw-audio-track=1&mw-buffer-duration=100",
        "connect": false,
        "audio-jitter": 0,
        "video-jitter": 0,
        "tally-preview": 0,
        "tally-program": 0,
        "audio-drop-frames": 0,
        "video-drop-frames": 0,
        "video-bit-rate": 0,
        "audio-bit-rate": 0,
        "video-width": 0,
        "video-height": 0,
        "video-scan": "progressive",
        "video-field-rate": false,
        "audio-num-channels": 0,
        "audio-sample-rate": 0,
        "audio-bit-count": 0
      }
    },
    {
      "position": 2,
      "connect": true,
      "device": {
        "device-name": "Pro Convert",
        "model": "NDI to HDMI",
        "product-id": 1040,
        "serial-no": "Z410221102013",
        "hw-revision": "Z",
        "fw-version": "1.1.916",
        "board-id": 2,
        "cpu-usage": 2.5,
        "memory-usage": 13.890000000000001,
        "core-temp": 45.0,
        "up-time": 25927,
        "output-state": "unconnected",
        "ip-addr": "169.254.4.59"
      }
    }
  ]
}
```

```

    },
    "ndi": {
      "name": "11",
      "url": "http://10.10.14.109:5588/bunny.mov?mw-bitrate=4096&mw-buffer-duration=600",
      "connect": false,
      "audio-jitter": 0,
      "video-jitter": 0,
      "tally-preview": 0,
      "tally-program": 0,
      "audio-drop-frames": 0,
      "video-drop-frames": 0,
      "video-bit-rate": 0,
      "audio-bit-rate": 0,
      "video-width": 0,
      "video-height": 0,
      "video-scan": "progressive",
      "video-field-rate": false,
      "audio-num-channels": 0,
      "audio-sample-rate": 0,
      "audio-bit-count": 0
    }
  },
  {
    "position": 3,
    "connect": true,
    "device": {
      "device-name": "NJ-play001",
      "model": "NDI to AIO",
      "product-id": 1057,
      "serial-no": "Z421230224001",
      "hw-revision": "Z",
      "fw-version": "1.1.916",
      "board-id": 3,
      "cpu-usage": 5.0,
      "memory-usage": 6.5800000000000001,
      "core-temp": 47.0,
      "up-time": 25926,
      "output-state": "connected",
      "ip-addr": "169.254.6.106"
    },
    "ndi": {
      "name": "SRT-Caller",
      "url": "srt://47.122.24.9:8000?streamid=play.srt/live/21&mode=caller&latency=125&mw-audio-track=1&mw-buffer-duration=60",
      "connect": false,
      "audio-jitter": 0,
      "video-jitter": 0,
      "tally-preview": 0,
      "tally-program": 0,
      "audio-drop-frames": 0,
      "video-drop-frames": 0,
      "video-bit-rate": 0,
      "audio-bit-rate": 0,
      "video-width": 0,
      "video-height": 0,
      "video-scan": "progressive",
      "video-field-rate": false,
      "audio-num-channels": 0,
      "audio-sample-rate": 0,
      "audio-bit-count": 0
    }
  },

```

```

{
  "position": 4,
  "connect": true,
  "device": {
    "device-name": "NJ-play004",
    "model": "NDI to AIO",
    "product-id": 1057,
    "serial-no": "Z421230224002",
    "hw-revision": "Z",
    "fw-version": "1.1.916",
    "board-id": 4,
    "cpu-usage": 26.32,
    "memory-usage": 10.1,
    "core-temp": 50.0,
    "up-time": 25927,
    "output-state": "connected",
    "ip-addr": "10.10.6.59"
  },
  "ndi": {
    "name": "MAC-MINI-2.LOCAL (Scan Converter)",
    "url": "",
    "connect": true,
    "audio-jitter": 100,
    "video-jitter": 100,
    "tally-preview": 0,
    "tally-program": 0,
    "audio-drop-frames": 2944,
    "video-drop-frames": 0,
    "video-bit-rate": 52322,
    "audio-bit-rate": 2610,
    "video-width": 3840,
    "video-height": 1212,
    "video-scan": "progressive",
    "video-field-rate": true,
    "audio-num-channels": 2,
    "audio-sample-rate": 48000,
    "audio-bit-count": 16
  }
},
{
  "position": 5,
  "connect": true,
  "device": {
    "device-name": "NJ-play05",
    "model": "NDI to HDMI",
    "product-id": 1040,
    "serial-no": "Z410221102009",
    "hw-revision": "Z",
    "fw-version": "1.1.918",
    "board-id": 5,
    "cpu-usage": 0.0,
    "memory-usage": 12.93,
    "core-temp": 40.0,
    "up-time": 25929,
    "output-state": "unconnected",
    "ip-addr": "169.254.5.73"
  },
  "ndi": {
    "name": "SRT Stream",
    "url": "srt://47.122.24.9:8000?streamid=play.srt/live/43&mode=caller&latency=125&mw-audio-track=1&mw-buffer-duration=100",
    "connect": false,

```

```

    "audio-jitter": 0,
    "video-jitter": 0,
    "tally-preview": 0,
    "tally-program": 0,
    "audio-drop-frames": 0,
    "video-drop-frames": 0,
    "video-bit-rate": 0,
    "audio-bit-rate": 0,
    "video-width": 0,
    "video-height": 0,
    "video-scan": "progressive",
    "video-field-rate": false,
    "audio-num-channels": 0,
    "audio-sample-rate": 0,
    "audio-bit-count": 0
  }
},
{
  "position": 6,
  "connect": true,
  "device": {
    "device-name": "Pro Convert",
    "model": "NDI to HDMI",
    "product-id": 1040,
    "serial-no": "Z410221102012",
    "hw-revision": "Z",
    "fw-version": "1.1.916",
    "board-id": 6,
    "cpu-usage": 46.149999999999999,
    "memory-usage": 21.02,
    "core-temp": 44.0,
    "up-time": 25925,
    "output-state": "unconnected",
    "ip-addr": "10.10.12.48"
  },
  "ndi": {
    "name": "MAC-MINI-2.LOCAL (Scan Converter)",
    "url": "",
    "connect": true,
    "audio-jitter": 100,
    "video-jitter": 100,
    "tally-preview": 0,
    "tally-program": 0,
    "audio-drop-frames": 1344,
    "video-drop-frames": 0,
    "video-bit-rate": 35911,
    "audio-bit-rate": 2211,
    "video-width": 3840,
    "video-height": 1212,
    "video-scan": "progressive",
    "video-field-rate": true,
    "audio-num-channels": 2,
    "audio-sample-rate": 48000,
    "audio-bit-count": 16
  }
},
{
  "position": 7,
  "connect": true,
  "device": {
    "device-name": "Pro Convert",
    "model": "HDMI 4K Plus",

```

```

    "product-id": 1025,
    "serial-no": "Z401230327003",
    "hw-revision": "Z",
    "fw-version": "1.1.272",
    "board-id": 7,
    "cpu-usage": 0.0,
    "memory-usage": 59.670000000000002,
    "core-temp": 65.0,
    "up-time": 25932,
    "output-state": "unconnected",
    "ip-addr": "169.254.11.184"
  },
  "ndi": {
    "name": "#07 (Z401230327003)",
    "enable": false,
    "num-clients": 0,
    "tally-preview": 0,
    "tally-program": 0,
    "audio-drop-frames": 0,
    "video-drop-frames": 0,
    "video-bit-rate": 0,
    "audio-bit-rate": 0,
    "video-width": 0,
    "video-height": 0,
    "video-scan": "progressive",
    "video-field-rate": false,
    "audio-num-channels": 0,
    "audio-sample-rate": 0,
    "audio-bit-count": 16
  }
},
{
  "position": 8,
  "connect": true,
  "device": {
    "device-name": "Pro Convert1",
    "model": "NDI to HDMI",
    "product-id": 1040,
    "serial-no": "Z410221102002",
    "hw-revision": "Z",
    "fw-version": "1.1.916",
    "board-id": 8,
    "cpu-usage": 2.5,
    "memory-usage": 12.710000000000001,
    "core-temp": 40.0,
    "up-time": 25925,
    "output-state": "unconnected",
    "ip-addr": "169.254.5.38"
  },
  "ndi": {
    "name": "RTSP",
    "url": "rtsp://10.10.11.54:554/mainstream?mw-tcp-first=1&mw-buffer-duration=500",
    "connect": false,
    "audio-jitter": 0,
    "video-jitter": 0,
    "tally-preview": 0,
    "tally-program": 0,
    "audio-drop-frames": 0,
    "video-drop-frames": 0,
    "video-bit-rate": 0,
    "audio-bit-rate": 0,
    "video-width": 0,

```

```

        "video-height": 0,
        "video-scan": "progressive",
        "video-field-rate": false,
        "audio-num-channels": 0,
        "audio-sample-rate": 0,
        "audio-bit-count": 0
    }
},
{
    "position": 9,
    "connect": true,
    "device": {
        "device-name": "Pro Convert",
        "model": "NDI to HDMI",
        "product-id": 1040,
        "serial-no": "Z410221102017",
        "hw-revision": "Z",
        "fw-version": "1.1.918",
        "board-id": 9,
        "cpu-usage": 0.0,
        "memory-usage": 12.550000000000001,
        "core-temp": 45.0,
        "up-time": 25925,
        "output-state": "unconnected",
        "ip-addr": "169.254.8.66"
    },
    "ndi": {
        "name": "None",
        "url": "",
        "connect": false,
        "audio-jitter": 0,
        "video-jitter": 0,
        "tally-preview": 0,
        "tally-program": 0,
        "audio-drop-frames": 0,
        "video-drop-frames": 0,
        "video-bit-rate": 0,
        "audio-bit-rate": 0,
        "video-width": 0,
        "video-height": 0,
        "video-scan": "progressive",
        "video-field-rate": false,
        "audio-num-channels": 0,
        "audio-sample-rate": 0,
        "audio-bit-count": 0
    }
},
{
    "position": 10,
    "connect": true,
    "device": {
        "device-name": "Pro Convert",
        "model": "NDI to HDMI",
        "product-id": 1040,
        "serial-no": "Z410221102016",
        "hw-revision": "Z",
        "fw-version": "1.1.918",
        "board-id": 10,
        "cpu-usage": 2.5,
        "memory-usage": 12.529999999999999,
        "core-temp": 45.0,
        "up-time": 25925,

```



```
    "output-state": "unconnected",
    "ip-addr": "169.254.8.61"
  },
  "ndi": {
    "name": "None",
    "url": "",
    "connect": false,
    "audio-jitter": 0,
    "video-jitter": 0,
    "tally-preview": 0,
    "tally-program": 0,
    "audio-drop-frames": 0,
    "video-drop-frames": 0,
    "video-bit-rate": 0,
    "audio-bit-rate": 0,
    "video-width": 0,
    "video-height": 0,
    "video-scan": "progressive",
    "video-field-rate": false,
    "audio-num-channels": 0,
    "audio-sample-rate": 0,
    "audio-bit-count": 0
  }
}
],
"status": 0,
"code": "Success"
}
```

5. 错误码

无接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

获取蜂鸣器状态

1.接口描述

接口说明：获取蜂鸣器当前状态

请求方式：POST [ip]/api/chassis/get-beeper

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 否 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|---------------|---------|------------------------------|
| status | Int | 状态码 |
| code | String | 状态描述 |
| beeper-enable | Boolean | 蜂鸣器是否开启 true: 是; false: 否 |

4.示例

获取蜂鸣器当前状态。

输入示例

```
None
```

输出示例

```
{
  "beeper-enable": false,
  "status": 0,
  "code": "Success"
}
```

5.错误码

无接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

设置蜂鸣器

1.接口描述

接口说明：设置蜂鸣器，当蜂鸣器状态为true时，若有警报，蜂鸣器会发声；当蜂鸣器状态为false时，即使有警报，蜂鸣器也不会发声。

请求方式：POST [ip]/api/chassis/set-beeper

| 管理员权限 | 登录认证 |
|-------|------|
| 否 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|---------------|----|---------|---------------------------|
| beeper-enable | 是 | Boolean | 是否开启蜂鸣器 true：是；false：否 |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

设置蜂鸣器。

输入示例

```
{
  "beeper-enable": false
}
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-----------------------|------|
| 7 | MW_STATUS_INVALID_ARG | 参数缺失 |

获取电源模块状态

1.接口描述

接口说明：获取电源板的电源模块状态

请求方式：POST [ip]/api/chassis/get-power

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 否 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|----------|--------|--|
| status | Int | 状态码 |
| code | String | 状态描述 |
| power-on | Int | 电源模块状态 0: 待机状态 1: 开机中 2: 已开机 3: 正在进入待机状态 |

4.示例

获取电源板的电源模块状态。

输入示例

```
None
```

输出示例

```
{
  "power-on": 2,
  "status": 0,
  "code": "Success"
}
```

5.错误码

无接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

设置电源模块

1.接口描述

接口说明：设置电源模块的状态。

请求方式：POST [ip]/api/chassis/set-power

| 管理员权限 | 登录认证 |
|-------|------|
| 是 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|----------|----|---------|-----------------------------|
| power-on | 是 | Boolean | 是否开启电源 true: 是; false: 否 |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

设置电源模块开启。

输入示例

```
{
  "power-on": true
}
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-----------------------|--------------|
| 1 | MW_STATUS_PENDING | 挂起待处理 |
| 7 | MW_STATUS_INVALID_ARG | 参数缺失 |
| 12 | MW_STATUS_DEVICE_LOST | 主机与电源控制板通信失败 |

获取触摸屏休眠模式

1.接口描述

接口说明：获取触摸屏休眠模式，当触摸屏休眠模式为true时，1分钟内未触摸，则会进入休眠状态；当触摸屏休眠模式为false时，触摸屏常亮。

请求方式：POST [ip]/api/chassis/get-screen

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 否 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------------|---------|----------------------------------|
| status | Int | 状态码 |
| code | String | 状态描述 |
| screen-sleep | Boolean | 触摸屏休眠模式是否开启 true: 是; false: 否 |

4.示例

获取触摸屏休眠模式是否开启。

输入示例

None

输出示例

```
{
  "screen-sleep": false,
  "status": 0,
  "code": "Success"
}
```

5.错误码

无接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

设置触摸屏休眠状态

1.接口描述

接口说明：设置触摸屏是否休眠，当触摸屏休眠模式为true时，1分钟内未触摸，则会进入休眠状态；当触摸屏休眠模式为false时，触摸屏常亮。

请求方式：POST [ip]/api/chassis/set-screen

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 否 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|--------------|----|---------|--------------------------------|
| screen-sleep | 是 | Boolean | 是否开启触摸屏休眠模式 true：是； false：否 |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

设置触摸屏休眠。

输入示例

```
{
  "screen-sleep": true
}
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-----------------------|------|
| 7 | MW_STATUS_INVALID_ARG | 参数缺失 |

定时自动重启设备

1.接口描述

接口说明：设置定时自动重启设备。

请求方式：POST [ip]/api/system/auto-reboot

| 管理员权限 | 登录认证 |
|-------|------|
| 是 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|--------|----|---------|--|
| enable | 是 | Boolean | 是否启用自动重启 true: 是; false: 否 |
| hour | 是 | Int | 自动重启时间, 时 |
| min | 是 | Int | 自动重启时间, 分 |
| week | 是 | [Array] | 每周重复 1: 星期一; 2: 星期二; 3: 星期三; 4: 星期四; 5: 星期五; 6: 星期六; 7: 星期天 |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

启用自动重启。

输入示例

```
{
  "enable":true,
  "hour":23,
  "min":59,
  "week":[1,2]
}
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-----------------------|---------|
| 7 | MW_STATUS_INVALID_ARG | 参数缺失或错误 |

获取设备信息

1.接口描述

接口说明：获取设备信息。请注意capability各子项是否为true，当true时相应的API访问才有效。

请求方式：POST [ip]/api/system/device-info

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 否 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|---------------|--------------------------------|----------|
| status | Int | 状态码 |
| code | String | 状态描述 |
| device-name | String | 设备名称 |
| product-id | String | 设备ID |
| product-name | String | 产品系列名称 |
| hardware-rev | String | 硬件版本 |
| serial-number | String | 设备序列号 |
| firmware-ver | String | 设备固件版本 |
| firmware-name | String | 设备固件名称 |
| build-time | String | 设备固件构建时间 |
| capability | CapabilityInfo | 能力信息 |

CapabilityInfo

| 参数名称 | 类型 | 描述 |
|------------------------|---------|---|
| support-timezone | Boolean | 支持时区 true: 是; false: 否 |
| support-ntp | Boolean | 支持NTP true: 是; false: 否 |
| support-station | Boolean | 支持Wi-Fi模式 true: 是; false: 否 |
| support-ap | Boolean | 支持AP模式 true: 是; false: 否 |
| support-online-upgrade | Boolean | 支持在线升级 true: 是; false: 否 |
| support-sc-control | Boolean | 支持Control Hub云管理 true: 是; false: 否 |
| support-if-prio | Boolean | 支持网卡优先级控制 true: 是; false: 否 |
| support-usb-ncm | Boolean | 支持启用或禁用USB NET, 通过 重启设备 生效 true: 启用; false: 禁用 |
| support-wifi-mutex | Boolean | WiFi模块支持同时运行 AP 和 STA模式 (仅适用于USB Fusion) true: 是; false: 否 |
| support-ipv6 | Boolean | 支持IPv6 true: 是; false: 否 |

4. 示例

获取设备信息。

输入示例

None

输出示例

```
{
  "device-name": "Pro Router ONE",
  "product-id": "0x601",
  "product-name": "Pro Router ONE",
  "hardware-rev": "B",
  "serial-number": "0123456789",
  "firmware-ver": "0.9.210",
  "firmware-name": "Development",
  "build-time": "2023-04-14 06:48:13",
  "capability": {
    "support-usbc-name": false,
    "support-timezone": true,
    "support-ntp": true,
    "support-station": true,
    "support-ap": true,
    "support-online-upgrade": false,
    "support-sc-control": false,
    "support-if-prio": false,
    "support-usb-ncm": false,
    "support-wifi-mutex": false,
    "support-ipv6": true
  },
  "status": 0,
  "code": "Success"
}
```

5. 错误码

无接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

获取设备基本信息

1.接口描述

接口说明：获取设备基本信息，包括CPU、内存、上电时间等。

请求方式：POST [ip]/api/system/info

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 否 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|-------------|--------------------------------|-----------|
| status | Int | 状态码 |
| code | String | 状态描述 |
| device-name | String | 设备名称 |
| mdns-url | String | mDNS URL |
| uptime | Int | 开机时长，单位：s |
| cpu | CPUInfo | CPU信息 |
| mem | MemInfo | 内存信息 |
| datetime | DateTimeInfo | 日期和时间信息 |
| auto-reboot | AutoRebootInfo | 自动重启信息 |

CPUInfo

| 参数名称 | 类型 | 描述 |
|-------|-----|---------------|
| total | Int | CPU 总时间 |
| idle | Int | CPU 空闲时间 |
| usage | Int | CPU 使用率 x 100 |

MemInfo

| 参数名称 | 类型 | 描述 |
|-------|-----|--------------|
| total | Int | 系统总内存，单位：KB |
| avail | Int | 系统可用内存，单位：KB |

DateTimeInfo

| 参数名称 | 类型 | 描述 |
|-------------|---------|---------------------------------|
| cur-time | String | 系统时间, 格式: yyyy-MM-dd HH:mm:ss |
| zonename | String | 时区名称 |
| ntp-enable | Boolean | 是否启用自动设置时间 true: 是; false: 否 |
| ntp-server1 | String | NTP服务器1 |
| ntp-server2 | String | NTP服务器2 |

AutoRebootInfo

| 参数名称 | 类型 | 描述 |
|--------|---------|--|
| enable | Boolean | 是否启用自动重启 true: 是; false: 否 |
| hour | Int | 自动重启时间, 时 |
| min | Int | 自动重启时间, 分 |
| week | [Array] | 每周重复 1: 星期一; 2: 星期二; 3: 星期三; 4: 星期四; 5: 星期五; 6: 星期六; 7: 星期天 |

4. 示例

获取设备基本信息。

输入示例

None

输出示例

```
{
  "device-name": "USB Fusion",
  "mdns-url": "xxxxx.local",
  "uptime": 8410,
  "cpu": {
    "total": 1624896,
    "idle": 1281701,
    "usage": 2110
  },
  "mem": {
    "total": 8069612,
    "avail": 7171768
  },
  "datetime": {
    "cur-time": "2021-12-20 13:25:57",
    "zonename": "Asia/Shanghai",
    "ntp-enable": true,
    "ntp-server1": "0.pool.ntp.org",
    "ntp-server2": "1.pool.ntp.org"
  },
  "auto-reboot": {
    "enable": true,
    "hour": 23,
    "min": 59,
    "week": [
      1,
      2
    ]
  },
  "status": 0,
  "code": "Success"
}
```

5. 错误码

无接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

设置日期和时间

1.接口描述

接口说明：设置日期和时间。

请求方式：POST [ip]/api/system/set-date-time

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|-------------|----|---------|--|
| ntp-enable | 是 | Boolean | 是否启用自动设置时间 true: 是; false: 否 |
| ntp-server1 | 是 | String | NTP服务器1 |
| ntp-server2 | 是 | String | NTP服务器2 |
| time | 否 | String | 时间, 格式: yyyy-MM-dd HH:mm:ss, ntp-enable为false时必填 |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

手动设置日期和时间。

输入示例

```
{
  "ntp-enable": false,
  "ntp-server1": "0.pool.ntp.org",
  "ntp-server2": "1.pool.ntp.org",
  "time": "2024-04-11 09:50:18"
}
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码, 其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-----------------------|---------|
| 7 | MW_STATUS_INVALID_ARG | 参数缺失或错误 |

设置设备名称

1.接口描述

接口说明：设置设备名称。

请求方式：POST [ip]/api/system/set-device-name

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|------|----|--------|------|
| name | 是 | String | 设备名称 |

3.输出参数

| 参数名称 | 类型 | 描述 |
|-----------------|---------|----------------------------|
| status | Int | 状态码 |
| code | String | 状态描述 |
| ext-need-reboot | Boolean | 是否需要重启设备 true：是 false：否 |

4.示例

设置设备名称为“Magewell-1”。

输入示例

```
{
  "name": "Magewell-1"
}
```

输出示例

```
{
  "status": 0,
  "code": "SUCCESS",
  "ext-need-reboot": true
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-----------------------|---------|
| 7 | MW_STATUS_INVALID_ARG | 参数缺失或错误 |

设置时区

1.接口描述

接口说明：设置时区。

请求方式：POST [ip]/api/system/timezone-set

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|----------|----|--------|------|
| zonename | 是 | String | 时区名称 |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

设置时区。

输入示例

```
{
  "zonename": "Asia/Shanghai"
}
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-----------------------|-------|
| 7 | MW_STATUS_INVALID_ARG | 参数缺失 |
| 16 | MW_STATUS_NOT_EXIST | 时区不存在 |

获取DNS

从2024年9月起弃用。

1.接口描述

接口说明：获取DNS。

请求方式：POST [ip]/api/network/get-dns

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 否 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|-----------|---------|--------------------------------|
| status | Int | 状态码 |
| code | String | 状态描述 |
| is-manual | Boolean | 是否手动设置DNS true: 是; false: 否 |
| dns1 | String | 首选DNS, 空字符表示未设置 |
| dns2 | String | 备用DNS, 空字符表示未设置 |

4.示例

获取DNS。

输入示例

None

输出示例

```
{
  "is-manual": false,
  "dns1": "10.0.1.3",
  "dns2": "",
  "status": 0,
  "code": "Success"
}
```

5.错误码

无接口业务逻辑相关的错误码, 其他错误码详见 [公共错误码](#)。

获取网卡信息

1.接口描述

接口说明：获取网卡信息。

请求方式：POST [ip]/api/network/if-info

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 否 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|------------------|----------------------------------|------------|
| status | Int | 状态码 |
| code | String | 状态描述 |
| device-name | String | 设备名称 |
| net | Array of NetData | 网卡列表 |
| ext-mobile-first | Boolean | 预留参数, 暂不支持 |

NetData

| 参数名称 | 类型 | 描述 |
|----------------|---------|--|
| support-enable | Boolean | 是否支持启停Wi-Fi/AP true: 是; false: 否 |
| enable | Boolean | 网卡服务是否开启 true: 是; false: 否 |
| prio | Int | 网卡优先级 0: 没有优先级, 一般指USB, ETH CONSOLE; 正常 1 ~ 99 |
| iface | String | 网卡名称 |
| type | Int | 网卡类型 0: 以太网 1: 无线网卡(Wi-Fi/AP) 2: USB Sharing 3: USB NET 4: 内置4G/5G 5: Bridge |
| use-dhcp | Boolean | 是否启用DHCP true: 是; false: 否 |
| ipaddr | String | IP 地址 |
| ipv6addr | String | IPv6 地址信息 |
| netmask | String | 子网掩码 |
| gateway | String | 网关地址 |
| dns1 | String | 首选DNS, 空字符表示未设置 |
| dns2 | String | 备用DNS, 空字符表示未设置 |
| mac | String | MAC 地址 |
| link-speed | Int | 速率 10: 10Mbps 100: 100Mbps 1000: 1Gbps 2500: 2.5Gbps 10000: 10Gbps 12: full-speed 480: high-speed 5000: super-speed-5g 10000: super-speed-10g |
| link-state | Int | 连接状态 0: 网口异常 1: 未连接 2: 已连接 |
| tx-speed-kbps | Int | 发送速度 (Kbps) |
| rx-speed-kbps | Int | 接收速度 (Kbps) |
| mode | Int | 无线网卡工作模式 0: STA 模式; 1: AP模式 |
| mode-lock | Boolean | 无线网卡工作模式是否锁定 true: 是; false: 否 |
| ssid | String | AP 网络名称 |
| reboot-require | Boolean | 无线网卡切换工作模式时, 是否需要重启 true: 是; false: 否 |

| 参数名称 | 类型 | 描述 |
|--------------|---------|--|
| enable-ncm | Boolean | USB NET是否被启用 true: 启用; false: 禁用 |
| sim-state | Int | SIM卡状态 0: not ready 1: ready 2: PIN 3: PUK |
| pin-remaind | Int | PIN 剩余次数 |
| standard | Int | 蜂窝网络信号类型 0: 未知; 2: 2G; 3: 3G; 4: 4G; 5: 5G 3G以下信号质量参考RSSI 4G/5G 信号参考RSRP |
| rsi | Int | 蜂窝网络信号强度 极好点: RSSI>-65dBm 好点: RSSI=-65 ~ -75dBm 中点: RSSI=-75 ~ -85dBm 差点: RSSI=-85 ~ -95dBm 极差点: RSSI<-95dBm |
| rsrp | Int | 4G/5G RSRP值 极好点: RSRP>-115dBm 好点: RSRP=-120 ~ -115dBm 中点: RSRP=-125 ~ -120dBm 差点: RSRP=-130 ~ -125dBm 极差点: RSRP<-130dBm |
| rsrq | Int | 4G/5G RSRQ值, 单位dB |
| sinr | Int | 4G/5G SINR值, 单位dB |
| imei-no | String | IMEI信息 |
| operator | String | 运营商 CHN-MOBLIE, CHN-UNICOM, CHN-CT, Orange, O2, Vodafone, AT&T, T-Mobile, Verizon, Google Fi |
| phone-number | String | SIM卡号码 |
| vendor | String | 4G/5G模块厂商信息 |
| product | String | 4G/5G模块型号信息 |

4.示例

获取网卡信息。

输入示例

None

输出示例

```
{
  "device-name": "00A601230913015",
  "ext-mobile-first": false,
  "net": [
    {
      "support-enable": false,
      "enable": true,
      "prio": 34,
      "iface": "eth0",
      "type": 0,
      "use-dhcp": true,
      "ipaddr": "10.10.6.222",
      "netmask": "255.255.240.0",
      "gateway": "10.10.0.1",
      "dns1": "10.10.1.3",
      "dns2": "",
      "ipv6addr": [
        "fe80::d2c8:57ff:fe81:c75e"
      ],
      "mac": "d0:c8:57:81:c7:5e",
      "link-speed": 1000,
      "link-state": 2,
      "tx-speed-kbps": 0,
      "rx-speed-kbps": 35
    }
  ]
  {
    "support-enable": true,
    "enable": true,
    "prio": 41,
    "iface": "wlan0",
    "type": 1,
    "mode": 1,
    "mode-lock": false,
    "ssid": "Magewell_ASR_3015_5G",
    "reboot-require": false,
    "use-dhcp": true,
    "ipaddr": "",
    "netmask": "",
    "gateway": "",
    "dns1": "",
    "dns2": "",
    "ipv6addr": [
      "fe80::d2c8:57ff:fe81:b7f1"
    ],
    "mac": "d0:c8:57:81:b7:f1",
    "link-speed": -1,
    "link-state": 2,
    "tx-speed-kbps": 0,
    "rx-speed-kbps": 0
  }
  {
    "support-enable": false,
    "enable": true,
    "prio": 36,
    "iface": "wwan0",
    "type": 4,
    "use-dhcp": true,
    "ipaddr": "",
    "netmask": "",
    "gateway": "",
  }
}
```

```
    "dns1": "",
    "dns2": "",
    "ipv6addr": [
    ],
    "mac": "1a:97:9f:84:3a:e2",
    "link-speed": -1,
    "link-state": 0,
    "tx-speed-kbps": 0,
    "rx-speed-kbps": 0,
    "enable-ipv6": false,
    "sim-state": 2,
    "pin-remaind": 3,
    "standard": 0,
    "rssi": 0,
    "rsrp": 0,
    "rsrq": 0,
    "sinr": 0,
    "band": "",
    "imei-no": "",
    "phone-number": "",
    "operator": "",
    "version": "",
    "vendor": "Quectel Wireless Solutions Co., Ltd.",
    "product": "EC20\EC25\EM05-CE LTE modem"
  }
],
"status": 0,
"code": "Success"
}
```

5. 错误码

无接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

配置以太网

1.接口描述

接口说明：配置以太网。

请求方式：POST [ip]/api/network/if-set

| 管理员权限 | 登录认证 |
|-------|------|
| 是 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|----------|----|---------|--|
| iface | 是 | String | 网卡名称, eth0 |
| use-dhcp | 是 | Boolean | 是否使用DHCP获取IP地址 true: 是否; false: 否, 手动配置 |
| ipaddr | 否 | String | IP 地址, 当 use-dhcp 为 false 时必填 |
| netmask | 否 | String | 子网掩码, 当 use-dhcp 为 false 时必填 |
| gateway | 否 | String | 网关地址, 当 use-dhcp 为 false 时必填 |
| dns1 | 否 | String | 首选DNS |
| dns2 | 否 | String | 备用DNS |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

设置以太网IP地址。

输入示例

```
{
  "iface": "eth0",
  "use-dhcp": false,
  "ipaddr": "10.10.10.88",
  "netmask": "255.255.240.0",
  "gateway": "10.10.0.1",
  "dns1": "10.10.0.3",
  "dns2": "",
}
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```


5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-------------------------|------------|
| 7 | MW_STATUS_INVALID_ARG | 参数缺失 |
| 31 | MW_STATUS_NOT_PERMITTED | 操作内容不是有线网卡 |

设置DNS

从2024年9月起弃用。

1.接口描述

接口说明：设置DNS。

请求方式：POST [ip]/api/network/set-dns

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|-----------|----|---------|--------------------------------|
| is-manual | 否 | Boolean | 是否手动设置DNS true: 是; false: 否 |
| dns1 | 否 | String | 首选DNS, 空字符表示未设置 |
| dns2 | 否 | String | 备用DNS, 空字符表示未设置 |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

设置DNS。

输入示例

```
{
  "is-manual": true,
  "dns1": "10.0.1.3",
  "dns2": ""
}
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

无接口业务逻辑相关的错误码, 其他错误码详见 [公共错误码](#)。

配置USB NET

1.接口描述

接口说明：配置USB NET。

请求方式：POST [ip]/api/network/usb-config

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|--------|----|--------|------------|
| iface | 是 | String | 网卡名称, usb0 |
| ipaddr | 是 | String | IP 地址 |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

配置USB NET。

输入示例

```
{
  "iface": "usb0",
  "ipaddr": "192.168.66.1"
}
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-----------------------|-----------------|
| 7 | MW_STATUS_INVALID_ARG | 参数缺失 |
| 12 | MW_STATUS_DEVICE_LOST | 操作内容不是USB NET网卡 |

清除升级状态

1.接口描述

接口说明：清除升级状态。

请求方式：POST [ip]/api/upgrade/clear

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

清除升级状态。

输入示例

```
None
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

无接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

获取固件版本和升级状态

1.接口描述

接口说明：获取当前固件版本信息和升级状态。

请求方式：POST [ip]/api/upgrade/state

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|------------------|--------|--|
| status | Int | 状态码 |
| code | String | 状态描述 |
| state | Int | 升级任务执行状态 0: 空闲 1: 初始化升级 2: 升级中 3: 升级完成 4: 固件在线下载中 |
| cur-ver | String | 当前固件版本号 |
| update-version | String | 最新固件版本号 |
| num-steps | Int | 升级任务总计需要几个步骤，仅在 state 为 2 状态下存在 |
| step | Int | 升级任务执行到第几步，仅在 state 为 2 状态下存在 |
| step-name | String | 当前执行步骤的名称，仅在 state 为 2 状态下存在 |
| step-progress | Int | 当前执行的步骤的进度，值为 0 ~ 100，单位%，仅在 state 为 2 状态下存在 |
| download-percent | Float | 固件在线下载百分比 |

4.示例

获取当前固件版本信息和升级状态。

输入示例

None

输出示例

```
{
  "status": 0,
  "code": "Success",
  "state": "updating",
  "cur-ver": "1.1.72",
  "update-version": "1.1.72",
  "num-steps": 4,
  "step": 2,
  "step-name": "Erasing image",
  "step-progress": 28
}
```

5. 错误码

无接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

执行固件升级操作

1.接口描述

接口说明：执行固件升级操作，升级过程中可以 [获取固件版本和升级状态](#)。

请求方式：POST [ip]/api/upgrade/update

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|-----------|----|---------|--------------------------------|
| is-online | 是 | Boolean | 是否为在线升级 true：是；false：否，手动升级 |
| mode | 是 | Int | 升级模式 0：Auto 模式 |
| timeout | 是 | Int | 升级进度持续不变时的超时时间，单位：s |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

执行固件升级操作

输入示例

```
{
  "is-online": false,
  "mode": 0,
  "timeout": 120
}
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-----------------------|-------|
| 7 | MW_STATUS_INVALID_ARG | 参数缺失 |
| 11 | MW_STATUS_DEVICE_BUSY | 正在升级 |
| 16 | MW_STATUS_NOT_EXIST | 文件不存在 |

上传固件

1.接口描述

接口说明：上传固件，上传文件格式必须为.mwf。

请求方式：POST [ip]/api/system/upload-fw

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

二进制封装的附件数据。

3.输出参数

| 参数名称 | 类型 | 描述 |
|-----------------|---------|-----------------------------------|
| status | Int | 状态码 |
| code | String | 状态描述 |
| need-clean-data | Boolean | 主版本更新，用户数据是否被清除 true：是；false：否 |
| up-to-date | Boolean | 是否是最新版本 true：是；false：否 |
| version | String | 上传的固件版本号 |
| size | Int | 上传的固件大小 |

4.示例

上传固件。

输入示例

```
-----WebKitFormBoundaryIQYf0LWb1KdjY6f3
Content-Disposition: form-data; name="file"; filename="usb_fusion_rev_a_2_5_0.mwf"
Content-Type: application/octet-stream

-----WebKitFormBoundaryIQYf0LWb1KdjY6f3--
```

输出示例

```
{
  "status": 0,
  "code": "Success",
  "need-clean-data": false,
  "up-to-date": false,
  "version": "2.5.0",
  "size": 258818308
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|---------------------------|------|
| 23 | MW_STATUS_FILE_TYPE_ERROR | 文件错误 |

添加用户

1.接口描述

接口说明：添加用户。

请求方式：POST [ip]/api/user/add

| 管理员权限 | 登录认证 |
|-------|------|
| 是 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|----------|----|--------|----------------------|
| username | 是 | String | 用户名 |
| password | 是 | String | 密码，密码明文的 sha256 哈希值。 |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

使用管理员账号添加用户

输入示例

```
{
  "username": "test",
  "password": "c4ca4238a0b923820dcc509a6f75849b"
}
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-----------------------|----------------|
| 7 | MW_STATUS_INVALID_ARG | 参数缺失 |
| 31 | NOT_PERMITTED | 非 Admin 用户，无权限 |
| 41 | CONSTRAINT_VIOLATION | 用户已存在 |

修改登录密码

1.接口描述

接口说明：用户修改自己的登录密码，修改时必须输入原密码。

请求方式：POST [ip]/api/user/ch-password

| 管理员权限 | 登录认证 |
|-------|------|
| 否 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|--------------|----|--------|-----------------------|
| password | 是 | String | 原密码，密码明文的 sha256 哈希值 |
| new-password | 是 | String | 新密码，密码明文的 sha256 哈希值。 |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

修改自己的登录密码

输入示例

```
{
  "password": "c4ca4238a0b923820dcc509a6f75849b",
  "new-password": "c1c224b03cd9bc7b6a86d77f5dace40191766c485cd55dc48caf9ac873335d6f"
}
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-----------------------|-------|
| 7 | MW_STATUS_INVALID_ARG | 参数缺失 |
| 39 | MISMATCH | 旧密码错误 |

删除用户

1.接口描述

接口说明：删除用户。

请求方式：POST [ip]/api/user/del

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|----------|----|--------|---|
| username | 是 | String | 用户名。此参数可通过 获取系统用户列表 获得。 |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

管理员删除用户

输入示例

```
{
  "username": "test"
}
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-----------------------|----------------|
| 7 | MW_STATUS_INVALID_ARG | 参数缺失 |
| 31 | NOT_PERMITTED | 非 Admin 用户，无权限 |

获取系统用户列表

1.接口描述

接口说明：获取系统用户列表信息。

请求方式：POST [ip]/api/user/get-all

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|-------------------------------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |
| users | Array of user | 用户列表 |

USER

| 参数名称 | 类型 | 描述 |
|----------|--------|---------------------|
| username | String | 用户名 |
| group | String | 用户所属分组，Admin 或 User |

4.示例

使用管理员账号获取用户列表

输入示例

None

输出示例

```
{
  "users": [
    {
      "username": "Admin",
      "group": "Admin"
    },
    {
      "username": "test",
      "group": "User"
    }
  ],
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|---------------|-----------------|
| 31 | NOT_PERMITTED | 非 Admin 用户, 无权限 |

用户登录

1.接口描述

接口说明：用户登录。登录成功后会在 Cookie 存放 Session ID (Cookie: sid-[serial number]=t2i704wbvoy51y408p588bpji010ibp0)。

请求方式：POST [ip]/api/user/login

| 管理员权限 | 登录认证 |
|-------|------|
| 否 | 否 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|----------|----|--------|---|
| username | 是 | String | 用户名。此参数可通过 获取系统用户列表 获得。 |
| password | 是 | String | 密码，密码明文的 sha256 哈希值。 |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |
| sid | String | 用户标识 |

4.示例

使用正确的用户名和密码登录

输入示例

```
{
  "username": "test",
  "password": "c4ca4238a0b923820dcc509a6f75849b"
}
```

输出示例

Response Headers

```
Content-Type: application/json; charset=utf-8
Expires: 0
Set-Cookie: sid-A506220808450=6440wa6u5fw8wv43f91v55cqkctnpv6
```

Response Body

```
{
  "status": 0,
  "code": "Success",
  "sid": "6440wa6u5fw8wv43f91v55cqkctnpv6"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-----------------------|-------|
| 7 | MW_STATUS_INVALID_ARG | 参数缺失 |
| 16 | MW_STATUS_NOT_EXIST | 用户不存在 |
| 36 | MW_STATUS_AUTH_FAILED | 密码错误 |

退出登录

1.接口描述

接口说明：用户退出登录，返回到登录界面。

请求方式：POST [ip]/api/user/logout

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 否 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

退出登录

输入示例

```
None
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-------------------------|----------|
| 37 | MW_STATUS_NOT_LOGGED_IN | 未登录，登录无效 |

重置用户密码

1.接口描述

接口说明：重置用户密码，无需输入原密码。

请求方式：POST [ip]/api/user/set-password

| 管理员权限 | 登录认证 |
|-------|------|
| 是 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|----------|----|--------|---|
| username | 是 | String | 用户名。此参数可通过 获取系统用户列表 获得。 |
| password | 是 | String | 新密码，密码明文的 sha256 哈希值。 |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

使用管理员账号修改 test 用户密码

输入示例

```
{
  "username": "test",
  "password": "c4ca4238a0b923820dcc509a6f75849b"
}
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-----------------------|----------------|
| 7 | MW_STATUS_INVALID_ARG | 参数缺失 |
| 16 | NOT_EXIST | 用户不存在 |
| 31 | NOT_PERMITTED | 非 Admin 用户，无权限 |

获取Token

1.接口描述

接口说明：获取用户Token。

请求方式：POST [ip]/api/user/token-all

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|------------------------------------|---------|
| status | Int | 状态码 |
| code | String | 状态描述 |
| tokens | Array of TokenInfo | Token列表 |

TokenInfo

| 参数名称 | 类型 | 描述 |
|----------|--------|---|
| username | string | 用户名 |
| token | string | Token |
| expire | int | Token失效时间, 从1970-01-01 00:00:00 +0000 (UTC)算起, -1为无限制, 单位: 秒。 |

4.示例

获取用户Token。

输入示例

None

输出示例

```
{
  "tokens": [
    {
      "username": "Admin",
      "token": "HTTrkb0UWh3KC157aYmSLHT3qDyswncF4ynAGJatr8i51aiZZbUT83MnhRJ0lghFB",
      "expire": -1
    }
  ],
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|---------------|-----------------|
| 31 | NOT_PERMITTED | 非 Admin 用户, 无权限 |

添加Token

1.接口描述

接口说明：在用户下添加token。

请求方式：POST [ip]/api/user/token-add

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 否 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|----------|----|--------|--|
| username | 是 | String | 用户名 |
| token | 是 | String | Token |
| expire | 是 | Int | Token失效时间, 从1970-01-01 00:00:00 +0000 (UTC)算起, -1为无限制, 单位: 秒 |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

添加Token。

输入示例

```
{
  "username": "Admin",
  "token": "HTrkb0UWh3KC157aYmSLHT3qDyswncF4ynAGJatr8i51aiZZbUT83MnhRJ0lghFB",
  "expire": -1
}
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-----------------------|----------------|
| 7 | MW_STATUS_INVALID_ARG | 参数缺失 |
| 31 | NOT_PERMITTED | 非 Admin 用户，无权限 |
| 41 | CONSTRAINT_VIOLATION | 用户已存在 |

删除Token

1.接口描述

接口说明：删除用户下Token。

请求方式：POST [ip]/api/user/token-del

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|-------|----|--------|-------|
| token | 是 | string | Token |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

删除用户下Token。

输入示例

```
{
  "token": "HTrkb0UWh3KC157aYmSLHT3qDyswncF4ynAGJatr8i51aiZZbUT83MnhRJ0lghFB"
}
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-----------------------|------|
| 7 | MW_STATUS_INVALID_ARG | 参数缺失 |

清除日志

1.接口描述

接口说明：清除全部系统日志。

请求方式：POST [ip]/api/log/clear

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

清除全部系统日志。

输入示例

```
None
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

无接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

导出日志

1.接口描述

接口说明：导出设备当前的系统日志，导出文件为 html 格式。

请求方式：POST [ip]/api/log/export

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|----------|----|--------|---------|
| filename | 是 | String | 导出的文件名称 |

3.输出参数

直接下载 html 格式日志文件到本地。

4.示例

导出设备当前的系统日志，文件命名为 Log_2024_04_11_14_01_08.html。

输入示例

```
{
  "filename": "Log_2024_04_11_14_01_08.html"
}
```

输出示例

直接下载 html 格式日志文件到本地。

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-----------------------|------|
| 7 | MW_STATUS_INVALID_ARG | 参数缺失 |

筛选日志

1.接口描述

接口说明：筛选日志。

请求方式：POST [ip]/api/log/filter

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|-------|----|--------|---|
| types | 是 | Int | 日志类型，当有多个类型时用英文逗号“,”隔开 all：全部；info：信息；warn：警告；error：错误 |
| key | 是 | String | 筛选关键字，可以为空字符串 |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|----------------------------------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |
| logs | Array of LogList | 日志列表 |

LogList

| 参数名称 | 类型 | 描述 |
|---------|--------|-----------------------------------|
| no | Int | 编号 |
| time | String | 日期时间 |
| type | String | 日志类型， info：信息；warn：警告；error：错误 |
| message | String | 日志内容 |

4.示例

筛选全部日志。

输入示例

```
{
  "types": "all",
  "key": ""
}
```

输出示例

```
{
  "status": 0,
  "code": "Success",
  "logs": [
    {
      "no": 0,
      "time": "2022/09/09 16:11:07.920",
      "type": "info",
      "message": "xxxxxx"
    },
    {
      "no": 1,
      "time": "2022/09/09 16:11:04.721",
      "type": "info",
      "message": "xxxxxx"
    }
  ]
}
```

5. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-----------------------|------|
| 7 | MW_STATUS_INVALID_ARG | 参数缺失 |

重置设备鉴权

1.接口描述

接口说明：判断是否允许重置设备。

请求方式：GET/POST [ip]/api/factory-reset-permission

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 否 | 否 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------------|---------|----------------------------------|
| status | Int | 状态码 |
| code | String | 状态描述 |
| reset-enable | Boolean | 是否允许重置设备 true: 允许; false: 不允许 |

4.示例

判断是否允许重置设备。

输入示例

None

输出示例

```
{
  "status": 0,
  "code": "Success",
  "reset-enable": true
}
```

5.错误码

无接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

Ping 测试

1.接口描述

接口说明：判断设备是否可以访问。在 固件更新 、 重置设备 、 修改 IP 地址 等操作完成后，设备需要重启，可以通过该接口判断设备是否已经重启完成。

请求方式：GET/POST [ip]/api/ping

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 否 | 否 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

判断设备是否可以访问。

输入示例

```
None
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

无接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

重启设备

1.接口描述

接口说明：重启设备，重启后需要重新登录。重启过程需要几分钟时间，可以使用 [Ping 测试](#) 判断设备是否已经重启。

请求方式：GET/POST [ip]/api/reboot

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------------|--------|----------------|
| status | Int | 状态码 |
| code | String | 状态描述 |
| delay | Int | 延时多少秒执行重启，单位：s |
| estimate-sec | Int | 估算重启时间，单位：s |

4.示例

重启设备。

输入示例

None

输出示例

```
{
  "status": 0,
  "code": "Success",
  "delay": 5,
  "estimate-sec": 15
}
```

5.错误码

无接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

重置设备

1.接口描述

接口说明：重置设备，恢复出厂设置。

请求方式：GET/POST [ip]/api/factory-reset

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 否 | 否 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

重置设备。

输入示例

```
None
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-------------------------|-------|
| 31 | MW_STATUS_NOT_PERMITTED | 无操作权限 |

获取证书信息

1.接口描述

接口说明：获取证书信息。

请求方式：POST [ip]/api/certificate/info

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|-------------|---------------------------------|--------------------------------|
| status | Int | 状态码 |
| code | String | 状态描述 |
| enable | Boolean | 启用HTTPS访问 true: 是; false: 否 |
| certificate | CertificateInfo | 证书信息 |

CertificateInfo

| 参数名称 | 类型 | 描述 |
|--------------|--------|-----------------------|
| subject-name | String | 主体名称, 包含证书持有者的身份信息 |
| issuer-name | String | 颁发者名称, 包含颁发证书的机构的身份信息 |
| version | Int | 版本 |
| serial-num | String | 序列号 |
| expires | String | 有效期截止时间 |
| valid-before | String | 有效期开始时间 |
| public-key | String | 公钥 |
| algorithm | String | 签名算法 |
| type | String | 算法类型 |

4.示例

获取证书信息。

输入示例

None

输出示例

```
{
  "status": 0,
  "code": "Success",
  "enable": true,
  "certificate": {
    "subject-name": "/C=CN/ST=GuangDong/L=ShenZhen/O=bolin-av.com/CN=bolin-av.com/emailAddress=av@bolin-av.com",
    "issuer-name": "/C=CN/ST=GuangDong/L=ShenZhen/O=Bolin Technology Root CA",
    "version": 1,
    "serial-num": "C11B50824C53B27C",
    "expires": "Mar 25 02:06:25 3022 GMT",
    "valid-before": "Nov 22 02:06:25 2022 GMT",
    "public-key": "30818902818100ba1e3aff73f880f3bc219f3d6714edb2cf4a4b8fb072cdf55c5058903af7691eeeb4cae6aacb71486b6cb2001e14cb5b9113d52f8db666c87b65465a4a1204976390d33ad42de91597bcab511d6ca9c0b7e3dad4f7584420672102406605eb4a1dcbf9871f85ec412947b27648ee48b03d2af9e8b9f915f534bec99d4d6ed3d70203010001",
    "algorithm": "1.2.840.113549.1.1.11",
    "type": "EVP_PKEY_RSA"
  }
}
```

5. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-------------------------|------|
| 6 | MW_STATUS_UNKNOWN_ERROR | 未知错误 |

启用HTTPS访问

1.接口描述

接口说明：启用HTTPS访问，重启后生效。

请求方式：POST [ip]/api/certificate/enable

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|--------|----|---------|-----------------------------|
| enable | 是 | Boolean | 启用HTTPS访问 true：是；false：否 |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

启用HTTPS访问。

输入示例

```
{
  "enable": false
}
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-------------------------|------|
| 6 | MW_STATUS_UNKNOWN_ERROR | 未知错误 |
| 7 | MW_STATUS_INVALID_ARG | 参数缺失 |

导入SSL证书

1.接口描述

接口说明：导入SSL证书。

请求方式：POST [ip]/api/certificate/import

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

| 参数名称 | 必选 | 类型 | 描述 |
|-------------|----|--------|-------|
| certificate | 是 | String | 根证书 |
| private-key | 是 | String | 根证书私钥 |

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

导入SSL证书

输入示例

```
{
  "certificate": "-----BEGIN CERTIFICATE-----\nMIIC1DCCAbwCCQDBG1CCTFOyfDANBgkqhkiG9w0BAQsFADBXMQswCQYDVQQGEwJD\nTjESMBAGA1UECAwJR3VhbmdEb25nMREwDwYDVQQHDAhTaGVuWmh1bjEhMB8GA1UE\nnCgwYQm9saW4gVG9jaG5vbG9neSBSb290IENBMCAxDTIyMTEyMjAyMDYy\nNVoYDzMw\nMjIwMzI1MDIwNjI1WjCBGjELMAKGA1UEBHMCMQ04xEjAQBgNVBAgMCUd1Yw5nRG9u\nZzERMA8GA1UEBwwIU2h1b1poZW4xFTATBgNVBAoMDGJvbGluLWF2LmNvbTEVMBMG\n\nA1UEAwMYM9saW4tYXYuY29tMR4wHAYJKoZIhvcNAQkBFg9hdkBib2xpbi1hdi5j\n\nb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBALoe0v9z+IDzvCGfPwC7bLP\n\nnSkuPsHLN9VxQWJA692ke7rTK5qrLcUhrbLIAHhTLW5ET1S+NtmbIe2VGWkoSBJdj\n\nnNM61C3pFZe8q1EdbKnAt+Pa1PdYRCBnIQJAZgXrSh3L+YcfhexBKUeydkjuSLA9\n\nnKvnoufkV9TS+yZ1NbtPXAgMBAAEwDQYJKoZIhvcNAQELBQADggEBAEsWsAbM8fFb\n\n\nvDtdkkE3pmjeZq1xUVBdNHgryVFz019Uic7eV48SGhLyLZ5AQF6htIJ0Z70pn+2G\n\n\nMoGZCu7CRmP/tvFHxUmFE4GRxHJYyxXlej3t2ju+5qiI0o3iGAaeobk6+OR519D0\n\n\nnL+UvLrhb0wbE53oZwMgGuH0cN08zAmh56ySgDjtm1rXVZugtEgOFv00RVFwUpERA\n\n\nnXLG6HyyFYaZf4yBYFGk6N804QF8JBsx68vavH41LODCY9kFUagVUSzTGvrLx2p0R\n\n\nnroxsbYJfimRtRs3xLifZaC+pXke1jIkiFq8qyxzzybRf4KsdmhaWebgeXmb2H9M\n\n\nn7VzaytYC1/k=\n-----END CERTIFICATE-----\n",
  "private-key": "-----BEGIN RSA PRIVATE KEY-----\nMIICXQIBAAKBgQC6Hjr/c/iA87whnz1nFO2yz0pLj7ByzfVcUFiQ0vdpHu60yuaq\n\n\nny3FiA2yyAB4Uy1uRE9UvjbZmyHt1RlpKEgSXY5DT0tQt6RWXvKtRHwypwLfj2tT3\n\n\nnWEQgZyECQGYf60ody/mHH4XsQSlHsnZI7kiwPSr56Ln5FfU0vsmdTW7T1wIDAQAB\n\n\nnAoGBAKzN/rdKRPPt2cJZ6Jf6DYzu/EVV/11Ir0Rr8R7RMQAwUTAZp7vyspar7zsB\n\n\nn8uwZuEXmZS0G5dFe2Xs5rAFT41/aVFEb+TYxpdDUx+k3GpveUhbX04qMYtWw4za\n\n\n\nnjzuKZSS2C6MdtOgayxSis+inrfRwPhV8mceia8yT6XiAD70JAKEA6Hpz1MFs2Ujx\n\n\n\nnYj8X05eHT+jmxARLCXKy/h6YmE5zPavPU5nCsVpg1Ej3N7oVy9mMxHyhR6iF+Cpb\n\n\n\nnKCKAjkGeewJBAMzy+eZ9kpcdD0U62DrPQvDecwQJ+gL28taUndMBAFjgG0ZjpDES\n\n\n\nnq9LCvZi75CBQvL/InMx42PMCGRrWkcc0D1UCQQDkLdRk/ZjdyDeMK8syFmdsaned\n\n\n\nn62zPEDJZNCSoUU/RN0JeluvACfojcV6c5tuf4p7E+GLAV8tuofo73xmjlTjAkAZ\n\n\n\nnA8MKpeMBfLM501BvBSMXgr74sPSJ7hg/po4I2S/WqSuA20KMFspQBCLeBHjYl0Ik\n\n\n\n\nnyliRAGMcbgwYw1h/fGtAkAcJf8uHBMHsb+Gz0SrQHs0ch0t/cu7kRUvno5bvsVC\n\n\n\n\nnX52nDmaC/QTfb9mFodkv3VvBHNOvG3vS/15YKWD8b5Pw\n\n\n\n\n-----END RSA PRIVATE KEY-----\n"
}
```

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-------------------------|---------|
| 7 | MW_STATUS_INVALID_ARG | 参数缺失 |
| 15 | MW_STATUS_WRITE_FAILED | IO 写入失败 |
| 40 | MW_STATUS_VERIFY_FAILED | 验证失败 |

删除SSL证书

1.接口描述

接口说明：删除SSL证书。

请求方式：POST [ip]/api/certificate/delete

| | |
|-------|------|
| 管理员权限 | 登录认证 |
| 是 | 是 |

2.输入参数

无

3.输出参数

| 参数名称 | 类型 | 描述 |
|--------|--------|------|
| status | Int | 状态码 |
| code | String | 状态描述 |

4.示例

删除SSL证书。

输入示例

None

输出示例

```
{
  "status": 0,
  "code": "Success"
}
```

5.错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码 | 定义 | 说明 |
|-----|-------------------------|------|
| 6 | MW_STATUS_UNKNOWN_ERROR | 未知错误 |