

USB Fusion

Overview

Introduction	1.1
API Agreement	1.2
API Status Codes	1.3
DEMO: Command Line Tool	1.4

Universal Interfaces

ping	2.1
reboot	2.2
factory-reset-permission	2.3
factory-reset	2.4

Dashboard

get-summary-info	3.1
set-hdmi-output-mode	3.2

BGM

add-bgm-to-list	4.1
bgm-next	4.2
bgm-play-control	4.3
bgm-previous	4.4
bgm-seek	4.5
bgm-set-policy	4.6
get-bgm-list	4.7
get-bgm-status	4.8
remove-bgm-from-list	4.9
sort-bgm-list	4.10
switch-bgm	4.11

Switching

create-scene-with-source	5.1
delete-scene	5.2
get-device-status	5.3
get-program-preview	5.4
get-scenes-list	5.5
get-source-thumbnail	5.6
get-video-player-status	5.7
rename-scene	5.8
seek-video-player	5.9
set-switch-action	5.10
set-video-player-status	5.11
switch-scene	5.12
upload-source-file	5.13

APP Settings

get-app-settings	6.1
get-encoder-params	6.2
set-app-password	6.3
set-encoder-format	6.4
set-network-port	6.5

System

device-info	7.1
info	7.2
set-device-name	7.3
timezone-set	7.4
set-date-time	7.5

User

login	8.1
logout	8.2
get-all	8.3
add	8.4
del	8.5
ch-password	8.6
set-password	8.7

Network

if-info	9.1
if-set	9.2
if-route	9.3
get-dns	9.4
set-dns	9.5
usb-config	9.6
wireless-mode	9.7
wireless-state	9.8
ap-passwd	9.9
ap-config	9.10
ap-get-config	9.11
wifi-scan	9.12
wifi-scan-results	9.13
wifi-connect	9.14
wifi-connect-auto	9.15
wifi-connect-state	9.16
wifi-disconnect	9.17
wifi-forget	9.18

Firmware Upgrade

online-check	10.1
online-check-result	10.2
upload-fw	10.3

update	10.4
state	10.5
clear	10.6

Log

clear	11.1
filter	11.2
export	11.3

WebSocket

MSG_WEB_SORKET_EVENT_CHANGE_INFO	12.1
--	------

Album

get-album-files-list	13.1
remove-album-file	13.2

Audio

get-volume-config	14.1
set-volume-config	14.2
get-audio-card-list	14.3
get-audio-card-mixer	14.4
get-mic-gain	14.5
get-mic-monitor	14.6
get-usb-audio-list	14.7
set-audio-card-mixer	14.8
set-mic-gain	14.9
set-mic-monitor	14.10
set-usb-audio	14.11

General

add-launch-scene	15.1
get-server-launch-scenes	15.2
get-server-settings	15.3
set-auto-switch	15.4
set-launch-scene	15.5
set-usb-mirror	15.6

Input

export-edid	16.1
get-edid-config	16.2
set-default-edid	16.3
set-edid-config	16.4
upload-edid	16.5
get-signal-info	16.6
get-video-config	16.7
set-video-config	16.8

get-webcam-config	16.9
set-webcam-config	16.10

Note

get-note-thumbnail	17.1
------------------------------------	------

Output

export-output-edid	18.1
get-hdmi-output-info	18.2
set-hdmi-output-mode	18.3
get-usb-output-config	18.4

Presentation

add-presentation	19.1
delete-presentation	19.2
get-presentations-list	19.3
rename-presentation	19.4
switch-presentation	19.5

Source

check-source-exist	20.1
check-source-used	20.2
delete-source	20.3
get-sources-list	20.4
import-source	20.5
rename-source	20.6
upload-source-file	20.7

Introduction

For USB Fusion, we have rich APIs for developers to interact with the device such as obtaining basic information about the device (device name, firmware version, etc.), modifying device configuration and upgrading firmware. These APIs are based on the Hyper Text Transfer Protocol (HTTP) and are lightweight and connectionless interfaces that response data in JavaScript Object Notation (JSON) format. This document gives you a detailed understanding of each API's functions and request mode.

APIs in this document apply to USB Fusion.

API Agreement

Overview

- Request protocol: HTTP
- Request mode: by default, GET and PUT are used to request data and commit, and POST is used to upload a file.
- Return data format: when the HTTP status code is 200, it returns JSON data, otherwise it returns HTTP error codes.
- Login authentication: carry sid=xxxxxxx in the Cookie.

Response Example

The JSON formatted data is as follows. The attribute of status refers to [API Status Codes](#). The status 0 indicates successful requests, otherwise the request is failed.

```
{
  status: 0,
  enable: true,
  enable-web-control: true
  ...
}
```

API Status Codes

```
{
  0: MW_STATUS_SUCCESS,
  1: MW_STATUS_PENDING,
  2: MW_STATUS_TIMEOUT,
  3: MW_STATUS_INTERRUPTED,
  4: MW_STATUS_TRY_AGAIN,
  5: MW_STATUS_NOT_IMPLEMENTED,
  6: MW_STATUS_UNKNOWN_ERROR,
  7: MW_STATUS_INVALID_ARG,
  8: MW_STATUS_NO_MEMORY,
  9: MW_STATUS_UNSUPPORTED,
  10: MW_STATUS_FILE_BUSY,
  11: MW_STATUS_DEVICE_BUSY,
  12: MW_STATUS_DEVICE_LOST,
  13: MW_STATUS_IO_FAILED,
  14: MW_STATUS_READ_FAILED,
  15: MW_STATUS_WRITE_FAILED,
  16: MW_STATUS_NOT_EXIST,
  17: MW_STATUS_TOO_MANY,
  18: MW_STATUS_TOO_LARGE,
  19: MW_STATUS_OVERFLOW,
  20: MW_STATUS_UNDERFLOW,
  21: MW_STATUS_FORMAT_ERROR,
  22: MW_STATUS_FILE_EXISTS,
  23: MW_STATUS_FILE_TYPE_ERROR,
  24: MW_STATUS_DEVICE_TYPE_ERROR,
  25: MW_STATUS_IS_DIRECTORY,
  26: MW_STATUS_READ_ONLY,
  27: MW_STATUS_RANGE_ERROR,
  28: MW_STATUS_BROKEN_PIPE,
  29: MW_STATUS_NO_SPACE,
  30: MW_STATUS_NOT_DIRECTORY,
  31: MW_STATUS_NOT_PERMITTED,
  32: MW_STATUS_BAD_ADDRESS,
  33: MW_STATUS_SEEK_ERROR,
  34: MW_STATUS_CROSS_DEVICE_LINK,
  35: MW_STATUS_NOT_INITIALIZED,
  36: MW_STATUS_AUTH_FAILED,
  37: MW_STATUS_NOT_LOGGED_IN,
  38: MW_STATUS_WRONG_STATE,
  39: MW_STATUS_MISMATCH,
  40: MW_STATUS_VERIFY_FAILED,
  41: MW_STATUS_CONSTRAINT_VIOLATION
  42: MW_STATUS_CANCELED,
    43: MW_STATUS_IN_PROGRESS,
    44: MW_STATUS_CONN_REFUSED,
    45: MW_STATUS_CONN_RESET,
    46: MW_STATUS_ADDR_IN_USE,
    47: MW_STATUS_NO_RESPONSE,
    48: MW_STATUS_INFO_CHANGED,
    49: MW_STATUS_INVALID_DATA,
    50: MW_STATUS_NEED_MORE_DATA,
    51: MW_STATUS_NO_BUFFER,
    52: MW_STATUS_BUFFER_TOO_SMALL,
    53: MW_STATUS_BUFFER_IS_EMPTY,
    54: MW_STATUS_BUFFER_IS_FULL
}
```

DEMO: Command Line Tool

To call USB Fusion API, wget and curl are supported in Linux, Windows, and Mac OS.

The location of cookie files varies according to the OS. Adjust the file path for your situation. The following examples are for Linux.

wget

1. Save your login information on cookies.

```
wget --save-cookies=/var/tmp/sid.txt --keep-session-cookies "http://192.168.66.1/mwapi?method=login&id=Admin&pass=e3afed0047b08059d0fada10f400c1e5" -q -O -
```

1. List all users.

```
wget --load-cookies=/var/tmp/sid.txt --keep-session-cookies "http://192.168.66.1/mwapi?method=get-users" -q -O -
```

1. Add a new user.

```
wget --load-cookies=/var/tmp/sid.txt --keep-session-cookies "http://192.168.66.1/mwapi?method=add-user&id=test&pass=c4ca4238a0b923820dcc509a6f75849b" -q -O -
```

curl

1. Save your login information on cookies.

```
curl --cookie-jar /var/tmp/sid.txt "http://192.168.66.1/mwapi?method=login&id=Admin&pass=e3afed0047b08059d0fada10f400c1e5"
```

1. List all users.

```
curl --cookie /var/tmp/sid.txt "http://192.168.66.1/mwapi?method=get-users"
```

1. Add a new user.

```
curl --cookie /var/tmp/sid.txt "http://192.168.66.1/mwapi?method=add-user&id=test&pass=c4ca4238a0b923820dcc509a6f75849b"
```


ping

Use the interface to detect whether the device is accessible without login.

This function is used to ensure that the device has restarted completely after `firmware update` , `reset all settings` or `change IP address` .

Request Mode

```
GET/POST /api/ping
```

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0: the device is ready. Refer to API Status Codes to find specific description for other values.

reboot

Use the interface to reboot the device as administrator and log in again after rebooting.

The reboot process may take a few minutes. You can use [ping](#) to determine whether the reboot is finished.

Request Mode

```
GET/POST /api/reboot
```

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully, and the device will reboot. Refer to API Status Codes to find specific description for other values.

factory-reset-permission

Use the interface to detect whether the device is allowed to be reset without login.

Request Mode

```
GET/POST /api/factory-reset-permission
```

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0: the device is ready. Refer to API Status Codes to find specific description for other values.

factory-reset

Use the interface to reset the device to default settings.

Request Mode

```
GET/POST /api/factory-reset
```

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0: the device is ready. Refer to API Status Codes to find specific description for other values.

get-summary-info

Use the interface to obtain status and parameters of the device, including basic information, Ethernet status, and USB Net status.

Request Mode

```
POST http://ip/mwapi/get-summary-info
```

Response Body

JSON structure is as follows:

```
{
  "status": 0,
  "device": {...},
  "wifi": {...},
  "ethernet": {...},
  "rndis": {...},
  "out": {...}
}
```

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

2. Basic Information (device {...})

```
"device": {
  "name": "USB Presenter",
  "model": "HDMI 2K Plus",
  "serial-no": "B401180706020",
  "hw-revision": "B",
  "fw-version": "1.1.72",
  "up-to-date": true,
  "input-hdmi-1-state": "1920x1080p60",
  "input-hdmi-2-state": "no-signal",
  "input-web-camera-state": "1280x720p30",
  "hdmi-output-mode": 0,
  "hdmi-output-state": "unconnected",
  "cpu-usage": 5.00,
  "memory-usage": 58.33,
  "core-temp": 46.76,
  "up-time": 8006,
  "storage-size": 0,
  "fan-rpm": 0
}
```

Name	Description
name	The name of the product family
model	The model name of the device, such as HDMI 2K Plus...
serial-no	The serial number of the device
hw-version	The hardware version of the device Value range: A - Z
fw-version	The current firmware version of the device
up-to-date	Whether the firmware is up to date If yes, it shows true; otherwise, it shows false.
input-hdmi-1-state	The status of the current input signal of the HDMI 1 port, including no-signal, locking, unsupported, or actual value (e.g. 1920x1080p60)

input-hdmi-2-state	The status of the current input signal of the HDMI 2 port, including no-signal, locking, unsupported, or actual value (e.g. 1920x1080p60)
input-web-camera-state	The status of the current input signal of the WEB CAMERA port, including no-signal, locking, unsupported, or actual value (e.g. 1920x1080p60)
hdmi-out-mode	The mode of the current output signal, including 0, 1 and 2, respectively representing loopthru HDMI 1, loopthru HDMI 2 and pgm
hdmi-out-state	The status of the current output signal, including unconnected, unsupported and active
cpu-usage	The current CPU usage
memory-usage	The current memory usage
core-temp	The current temperature (°C) of the device's processor
up-time	The elapsed time since the device's last boot-up, in seconds
storage-size	The storage capacity in MB
fan-rpm	The fan speed per minute

3. WiFi Status (wifi {...})

```
"wifi": {
  "state": "magewell-001",
  "mac-addr": "70:B3:D5:75:D2:41",
  "ip-addr": "0.0.0.0",
  "ip-mask": "0.0.0.0",
  "gw-addr": "0.0.0.0",
  "dns-addr": "0.0.0.0",
  "tx-speed-kbps": 0,
  "rx-speed-kbps": 0
}
```

Name	Description
state	The connection status of WiFi, including down and disconnected
mac-addr	The MAC address
ip-addr	The WiFi IP address
ip-mask	The subnet mask address
gw-addr	The gateway address
dns-addr	The DNS server address
tx-speed-kbps	The WiFi send speed in Kbps
rx-speed-kbps	The WiFi receive speed in Kbps

4. Ethernet Status (ethernet {...})

```
"ethernet": {
  "state": "disconnected",
  "mac-addr": "70:B3:D5:75:D2:41",
  "ip-addr": "0.0.0.0",
  "ip-mask": "0.0.0.0",
  "gw-addr": "0.0.0.0",
  "dns-addr": "0.0.0.0",
  "tx-speed-kbps": 0,
  "rx-speed-kbps": 0
}
```

Name	Description
state	The Ethernet connection status, including down, disconnected, 10m, 100m, 1000m, 2500m, 5000m, and 10000m
mac-addr	The MAC address
ip-addr	The Ethernet IP address
ip-mask	The subnet mask address
gw-addr	The gateway address
dns-addr	The DNS server address
tx-speed-kbps	The Ethernet send speed, in Kbps

rx-speed-kbps	The Ethernet receive speed, in Kbps
---------------	-------------------------------------

5. USB Net Status (rndis {...})

```
"rndis": {
  "state": "high-speed",
  "ip-addr": "192.168.66.1",
  "tx-speed-kbps": 0,
  "rx-speed-kbps": 0
}
```

Name	Description
state	The USB connection status, including disconnected, full-speed, high-speed, super-speed-5g, and super-speed-10g
ip-addr	The USB RNDIS address
tx-speed-kbps	The current send speed, in Kbps
rx-speed-kbps	The current receive speed, in Kbps

6. USB OUT Status (out {...})

```
"out": {
  "video-format": "NV12",
  "connection": "USB 3.0",
  "audio-drop-frames": 0,
  "video-drop-frames": 0,
  "video-width": 1920,
  "video-height": 1080,
  "video-scan": "progressive",
  "video-field-rate": 0.00,
  "audio-num-channels": 2,
  "audio-sample-rate": 48000,
  "audio-bit-count": 16
}
```

Name	Description
video-format	The output format of UVC video, including NV12, YUYV, MJPEG, etc.
connection	The connection status of USB, including USB 3.0, USB 2.0, and disable
audio-drop-frames	The number of audio frames dropped in the previous second
audio-num-channels	The total number of audio channels
audio-sample-rate	The audio sampling rate, including 32000, 44100, etc.
audio-bit-count	The audio bit rate, including 16, 20, 24, etc.
video-drop-frames	The number of video frames dropped in the previous second
video-bit-rate	The video encoding rate in the previous second, in Kbps
video-width	The width of the video and the total number of pixels
video-height	The height of the video and the total number of pixels
video-scan	The scan mode, including progressive, interlaced and psf
video-field-rate	The frame rate, including 24, 25, 29.97, 30, 48, 50, 59.94, and 60

set-hdmi-output-mode

Use the interface to set the output mode of HDMI.

Request Mode

```
POST http://ip/mwapi/set-hdmi-output-mode
```

Parameter	Description
mode	The mode type 0: LOOPTHRU HDMI 1 1: LOOPTHRU HDMI 2 2: PGM

Response Body

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

Name	Description
status	0: the setting is successful. Refer to API Status Codes to find specific description for other values.

add-bgm-to-list

Use the interface to add audio files to the current playlist.

Request Mode

```
POST http://ip/mwapi/add-bgm-to-list
```

Parameter	Description
id	The source ID

```
{  
  "id":203  
}
```

Response Body

JSON structure is as follows:

```
{  
  "message": "success",  
  "status": 0  
}
```

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

bgm-next

Use the interface to switch to the next BGM.

Request Mode

```
POST http://ip/mwapi/bgm-next
```

Response Body

JSON structure is as follows:

```
{  
  "message": "success",  
  "status": 0  
}
```

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

bgm-play-control

Use the interface to control the play of BGM.

Request Mode

```
POST http://ip/mwapi/bgm-play-control
```

Parameter	Description
playStatus	The play status 0: pause 1: playing

```
{  
  "playStatus":0  
}
```

Response Body

JSON structure is as follows:

```
{  
  "message": "success",  
  "status": 0  
}
```

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

bgm-previous

Use the interface to switch to the previous BGM.

Request Mode

```
POST http://ip/mwapi/bgm-previous
```

Response Body

JSON structure is as follows:

```
{  
  "message": "success",  
  "status": 0  
}
```

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

bgm-seek

Use the interface to seek the BGM to the specific progress.

Request Mode

```
POST http://ip/mwapi/bgm-seek
```

Parameter	Description
progress	Seek to the anchor point (0 - 1000000)

```
{  
  "progress":500000  
}
```

Response Body

JSON structure is as follows:

```
{  
  "message": "success",  
  "status": 0  
}
```

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

bgm-set-policy

Use the interface to set the play policy of BGM.

Request Mode

```
POST http://ip/mwapi?method=bgm-set-policy
```

Parameter	Description
musicPolicy	The play policy 1: play in order 2: shuffle the playlist 3: repeat a single song 4: repeat the playlist

```
{  
  "musicPolicy":1  
}
```

Response Body

JSON structure is as follows:

```
{  
  "message": "success",  
  "status": 0  
}
```

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

get-bgm-list

Use the interface to get the BGM list.

Request Mode

```
POST http://ip/mwapi/get-bgm-list
```

Response Body

JSON structure is as follows:

```
{
  "info": {
    "musicList": [{...}],
    "pageIndex": 0,
    "totalCount": 3
  },
  "message": "success",
  "result": 0
}
```

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.
pageIndex	Shows the page number: 0, 1, 2...
totalCount	Shows the total number of items

2. BGM List of Current Page (bgmList {...})

```
"bgmList":[
  {
    "artist": "Magewell",
    "duration": 231000,
    "id": 2006,
    "title": "Magewell 01"
  },
  {
    "artist": "Magewell",
    "duration": 194000,
    "id": 2038,
    "title": "Magewell 02"
  },
  {
    "artist": "Magewell",
    "duration": 240000,
    "id": 2037,
    "title": "Magewell 03"
  }
]
```

Name	Description
id	The BGM ID
title	The song name
artist	The artist
duration	The duration in ms

get-bgm-status

Use the interface to get the play status of BGM.

Request Mode

```
POST http://ip/mwapi/get-bgm-status
```

Response Body

JSON structure is as follows:

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

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.
musicStatus	The status of BGM

2. Current Status

```
"musicStatus": {
  "currentMusicId": 2038,
  "duration": 194000,
  "policy": 2,
  "progress": 500000,
  "status": 0
}
```

Name	Description
currentMusicId	The ID of current BGM
duration	The duration of audio, in ms
Status	The play status
policy	The play policy
progress	The play progress (0 - 1000000)

remove-bgm-from-list

Use the interface to remove a song from the BGM playlist.

Request Mode

```
POST http://ip/mwapi/remove-bgm-from-list
```

Name	Description
id	The source ID

```
{  
  "id":203  
}
```

Response Body

JSON structure is as follows:

```
{  
  "message": "success",  
  "status": 0  
}
```

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

sort-bgm-list

Use the interface to change the order of playlist.

Request Mode

```
POST http://ip/mwapi/sort-bgm-list
```

Parameter	Description
id	The BGM ID
position	The position after the movement (The subscript in the new list: 0, 1, 2...)

```
{  
  "id":203,"position":0  
}
```

Response Body

JSON structure is as follows:

```
{  
  "message": "success",  
  "status": 0  
}
```

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

switch-bgm

Use the interface to switch the BGM.

Request Mode

```
POST http://ip/mwapi/switch-bgm
```

Parameter	Description
id	BGM ID

```
{  
  "id":203  
}
```

Response Body

JSON structure is as follows:

```
{  
  "message": "success",  
  "status": 0  
}
```

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

create-scene-with-source

Use the interface to create a new scene containing the specific source.

Request Mode

POST http://ip/mwapi/get-scenes-list

Parameter	Description
sourceId	The source ID

Response Body

JSON structure is as follows:

```
{
  "status": 0,
  "message": "success",
  "info": {...}
}
```

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

2. New Scene Information (info {...})

```
"info": {
  "audioList": [
    {
      "layerId": 0,
      "muted": 0,
      "volume": 1000000
    }
  ],
  "dividerLine": {
    "color": 0,
    "width": 0,
    "x1": 500000,
    "x2": 500000,
    "y1": 0,
    "y2": 1000000
  },
  "id": 104,
  "layerList": [
    {
      "cropWindow": {
        "height": 1000000,
        "type": 1,
        "width": 1000000,
        "x": 0,
        "y": 0
      },
      "flip": 0,
      "frame": {
        "color": 16777215,
        "width": 0
      },
      "given": 0,
      "givenBg": 0,
      "rotation": 0,
    }
  ]
}
```

```

        "sourceDuration": 0,
        "sourceHeight": 1461,
        "sourceId": 4,
        "sourceType": 1,
        "sourceWidth": 2224,
        "srcWindow": {
            "height": 1000000,
            "width": 856262,
            "x": 71869,
            "y": 0
        },
        "window": {
            "height": 1000000,
            "width": 1000000,
            "x": 0,
            "y": 0
        }
    },
    {
        "cropWindow": {
            "height": 1000000,
            "type": 0,
            "width": 1000000,
            "x": 0,
            "y": 0
        },
        "flip": 0,
        "frame": {
            "color": 0,
            "width": 0
        },
        "givenBg": 0,
        "rotation": 0,
        "sourceHeight": 0,
        "sourceId": 0,
        "sourceType": 0,
        "sourceWidth": 0,
        "srcWindow": {
            "height": 1000000,
            "width": 1000000,
            "x": 0,
            "y": 0
        },
        "window": {
            "height": 0,
            "width": 0,
            "x": 0,
            "y": 0
        }
    }
],
"name": "default pic",
"template": 3,
"type": 3
}

```

Name	Description
id	The scene ID
name	The scene name
type	The scene type 4: note Others: scenes
template	The combination template 0: picture in picture 1: side by side 2: single-layer picture
dividerLine	The attribute of divider line
layerList	The layer list

2. Divider Line (dividerLine)

Name	Description
color	The color of divider line in uint32 (gbra)
x1	The x value of divider line top point
y1	The y value of divider line top point
x2	The x value of divider line end point
y2	The y value of divider line end point

3. Scene

Name	Description
sourceId	The source ID
sourceType	The source type 1: picture 2: video 3: HDMI 1 scene 4: HDMI 2 scene 5: USB-Camera 6: audio 7: background picture 8: automatic camera
sourceWidth	The width of the source picture
sourceHeight	The height of the source picture
sourceDuration	The length of the source video
mediaProgress	The play process of the source media, only for video
rotation	The angle of rotation which is in a counterclockwise direction Range: [0 - 360]
flip	The attribute of flip 0: not flip 1: flip horizontally 2: flip vertically 3: flip horizontally and vertically
frame	The frame attribute
window	The window attribute
srcWindow	The attribute of the content window
cropWindow	The attribute of the cropping window

4. Window (window)

Name	Description
x	x, relative value (0 - 1000000), the same below
y	y
width	Width
height	Height
type	(Optional) Aspect ratio 0: free ratio 1: original ratio 2: 4:3 3: 16:9 4: 1:1

delete-scene

Use the interface to delete a scene.

Request Mode

```
POST http://ip/mwapi/delete-scene
```

Parameter	Description
sceneIds	The scene ID Uses "," to separate multiple IDs.
showId	The presentation ID

```
{  
  "sceneIds": "100",  
  "showId": 7  
}
```

Response Body

JSON structure is as follows:

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

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

2. The scene to delete is being edited. (Other non-editing scenes in this request sequence will be deleted successfully.)

```
{  
  "info": {  
    "editSceneId": 112  
  },  
  "message": "device busy",  
  "status": 11  
}
```


get-device-status

Use the interface to get the status information of the device.

Request Mode

```
POST http://ip/mwapi/get-device-status
```

Response Body

JSON structure is as follows:

```
{
  "result": 0,
  "message": "success",
  "info": {...}
}
```

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

2. Basic Status Information (info {...})

```
"info": {
  "annotationStatus": {
    "arrowWidth": 4,
    "boardMode": 2,
    "eraserWidth": 34,
    "fillColor": 0,
    "highlighterWidth": 20,
    "inkDuration": 2000,
    "inkStyle": 0,
    "laserpointerPosX": 142089,
    "laserpointerPosY": 675347,
    "lineWidth": 2,
    "lowerFactor": 0,
    "shapeType": 4,
    "strokeColor": 44230911,
    "upperFactor": 500000,
    "useStroke": false
  },
  "ftbEnable": 0,
  "musicStatus": {
    "currentMusicId": 201,
    "duration": 253000,
    "policy": 4,
    "progress": 147919,
    "status": 1
  },
  "recordStatus": {
    "lastRecordError": 0,
    "recording": 1,
    "timeRecording": 18,
    "timeRemaining": 84640
  },
  "sceneStatus": {
    "cropping": 0,
    "croppingLayerId": 0,
    "currentID": 246,
    "currentShowId": 4,
    "editing": 0,
  }
}
```

```

    "currentShowId": 7,
    "defaultNoteBgId": 1001,
    "isFrozen": 0,
    "lastNoteId": 114,
    "lastSceneId": 246,
    "noteEditing": 0
  },
  "srtStatus": [
    {
      "port": 9110,
      "srtId": 0,
      "stoped": 0
    },
    {
      "port": 9111,
      "srtId": 1,
      "stoped": 0
    },
    {
      "port": 9112,
      "srtId": 2,
      "stoped": 0
    },
    {
      "port": 9113,
      "srtId": 3,
      "stoped": 0
    },
    {
      "port": 9114,
      "srtId": 4,
      "stoped": 0
    }
  ],
  "videoPlayerStatus": [
    {
      "layerId": 0,
      "playStatus": 0,
      "progress": 801324,
      "sourceDuration": 18875,
      "sourceId": 1191
    }
  ]
},

```

Name	Description
annotationStatus	The configuration of annotation
ftbEnable	Whether to enable FTB 0: not enabled 1: enabled
musicStatus	The status information of the BGM player
recordStatus	The status information of the recording
sceneStatus	The status information of the current scene
srtStatus	The SRT server status information of the captured scene
videoPlayerStatus	The video player status information of the current scene

Status Information of BGM Player (musicStatus)

Name	Description
currentMusicId	The ID of current music
duration	The duration of audio file in ms
policy	The play policy 1: play in order 2: shuffle the playlist 3: repeat a single song 4: repeat the playlist
progress	The play progress (0 - 1000000)

status	The play status 0: pause 1: playing
--------	---

Status Information of Recording (recordStatus)

Name	Description
lastRecordError	The information of the last record error
recording	Recording task status 0: not enabled 1: recording
timeRecording	The duration of the recording in seconds
timeRemaining	The remaining recording duration in seconds

Status Information of Current Scene (sceneStatus)

Name	Description
currentShowId	The ID of the current presentation
currentID	The ID of the current preview scene
lastNotelId	The ID of the last preview note
lastScenelId	The ID of the last preview scene
editing	Whether the scene is editing 0: false 1: true
editScenelId	The ID of the editing scene
defaultNoteBgId	The ID of the default note background source
cropping	Whether the layer is cropping 0: false 1: true
croppingLayerId	The ID of the cropping layer
isFrozen	Whether the preview scene is frozen 0: false 1: true

Video Player Status Information of Current Scene (videoPlayerStatus)

Name	Description
layerId	The layer ID
playStatus	The player status 0: pause 1: playing
progress	The player progress (0 - 1000000)
sourceDuration	The length of the source video in ms
sourceId	The source ID

SRT Streaming Configuration

Name	Description
srtId	The srtId 0: preview scene 0: HDMI 1 2: HDMI 2 3: USB Camera
port	The port number
stopped	The start or stop status 1: stopped 0: started

get-scenes-list

Use the interface to get the scene list of the current presentation.

Request Mode

```
POST http://ip/mwapi/get-scenes-list
```

Response Body

JSON structure is as follows:

```
{
  "status": 0,
  "info": {
    "totalCount": 3,
    "pageIndex": 0,
    "audioList": [{...}],
    "sceneList": [{...}]
  }
}
```

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.
totalCount	The total number of scene items

2. Presentation Audio Configuration (audioList [{...}])

```
"audioList": [
  {
    "muted": 0,
    "type": 1,
    "volume": 1000000
  },
  {
    "muted": 1,
    "type": 2,
    "volume": 1000000
  },
  {
    "muted": 0,
    "type": 3,
    "volume": 1000000
  },
  {
    "muted": 0,
    "type": 4,
    "volume": 1000000
  },
  {
    "muted": 0,
    "type": 5,
    "volume": 1000000
  },
  {
    "muted": 0,
    "type": 6,
    "volume": 1000000
  }
],
```

Name	Description
type	Audio type 1: output 2: monitor 3: mic 4: bgm 5: uac in 6: line in
muted	Muted 0: NO 1: YES
volume	Volume (0 - 1000000)

3. Scene List of Current Page (sceneList [{...}])

```

"sceneList":[
  {
    "audiolist": [
      {
        "layerId": 0,
        "muted": 0,
        "type": 256,
        "volume": 1000000
      }
    ],
    "dividerLine": {
      "color": 0,
      "width": 0,
      "x1": 500000,
      "x2": 500000,
      "y1": 0,
      "y2": 1000000
    },
    "id": 100,
    "layerList": [
      {
        "cropWindow": {
          "height": 1000000,
          "type": 0,
          "width": 1000000,
          "x": 0,
          "y": 0
        },
        "flip": 0,
        "frame": {
          "color": 0,
          "width": 0
        },
        "givenBg": 0,
        "mediaProgress": 0,
        "rotation": 360,
        "sourceDuration": 0,
        "sourceHeight": 1080,
        "sourceId": 1,
        "sourceThumbHeight": 144,
        "sourceThumbWidth": 256,
        "sourceType": 3,
        "sourceWidth": 1920,
        "srcWindow": {
          "height": 1000000,
          "width": 1000000,
          "x": 0,
          "y": 0
        },
        "window": {
          "height": 1000000,
          "width": 1000000,
          "x": 0,
          "y": 0
        }
      }
    ],
  }
]

```

```

        "cropWindow": {
            "height": 1000000,
            "type": 0,
            "width": 1000000,
            "x": 0,
            "y": 0
        },
        "flip": 0,
        "frame": {
            "color": 0,
            "width": 0
        },
        "givenBg": 0,
        "rotation": 0,
        "sourceHeight": 0,
        "sourceId": 0,
        "sourceThumbHeight": 0,
        "sourceThumbWidth": 0,
        "sourceType": 0,
        "sourceWidth": 0,
        "srcWindow": {
            "height": 1000000,
            "width": 1000000,
            "x": 0,
            "y": 0
        },
        "window": {
            "height": 0,
            "width": 0,
            "x": 0,
            "y": 0
        }
    }
},
"name": "",
"template": 2,
"type": 3
},
{
    "audiolist": [],
    "dividerLine": {
        "color": 0,
        "width": 0,
        "x1": 500000,
        "x2": 500000,
        "y1": 0,
        "y2": 1000000
    },
    "id": 101,
    "layerList": [
        {
            "cropWindow": {
                "height": 1000000,
                "type": 0,
                "width": 1000000,
                "x": 0,
                "y": 0
            },
            "flip": 0,
            "frame": {
                "color": 0,
                "width": 0
            },
            "givenBg": 0,
            "rotation": 0,
            "sourceHeight": 1080,
            "sourceId": 1001,
            "sourceThumbHeight": 0,
            "sourceThumbWidth": 0,
            "sourceType": 7,
            "sourceWidth": 1920,
            "srcWindow": {
                "height": 1000000,

```



```

        "width": 1000000,
        "x": 0,
        "y": 0
    },
    "thumbHeight": 144,
    "thumbWidth": 256,
    "window": {
        "height": 1000000,
        "width": 1000000,
        "x": 0,
        "y": 0
    }
},
{
    "cropWindow": {
        "height": 1000000,
        "type": 0,
        "width": 1000000,
        "x": 0,
        "y": 0
    },
    "flip": 0,
    "frame": {
        "color": 0,
        "width": 0
    },
    "givenBg": 0,
    "rotation": 0,
    "sourceDuration": 0,
    "sourceHeight": 1080,
    "sourceId": 2039,
    "sourceThumbHeight": 0,
    "sourceThumbWidth": 0,
    "sourceType": 1,
    "sourceWidth": 1920,
    "srcWindow": {
        "height": 1000000,
        "width": 1000000,
        "x": 0,
        "y": 0
    },
    "window": {
        "height": 1000000,
        "width": 1000000,
        "x": 0,
        "y": 0
    }
}
],
"name": "note",
"template": 0,
"type": 4
}
]

```

1. Scene

Name	Description
id	The scene ID
name	The scene name
type	The scene type 4: note Others: scenes (no detailed types)
template	The combination template 0: picture in picture 1: side by side 2: single-layer picture
dividerLine	The attribute of the divider line
layerList	The layer list

2. Audio Configuration of the Scene

Name	Description
type	Audio type 0x0100: HDMI 1 0x0200: video 0x0300: HDMI 2 0x0400: USB Camera 0x0500: picture
muted	Muted 0: NO 1: YES
volume	Volume: 0 - 1000000

3. Divider Line (dividerLine)

Name	Description
color	The color of divider line in uint32 (gbra)
x1	The x value of divider line top point
y1	The y value of divider line top point
x2	The x value of divider line end point
y2	The y value of divider line end point

4. Layer (layer)

Name	Description
sourceId	The source ID, accessing through http (<code>http://[deviceIP]/source/(sourceId)</code>)
sourceType	The source type 1: picture 2: video 3: HDMI 1 scene 4: HDMI 2 scene 5: USB-Camera 6: audio 7: background picture 8: automatic camera
sourceWidth	The width of the source picture
sourceHeight	The height of the source picture
sourceDuration	The length of the source video
mediaProgress	The play process of the source media, only for video
rotation	The rotation angle which is in a counterclockwise direction Range: [0 - 360]
flip	The flip attribute 0: not flip 1: flip horizontally 2: flip vertically 3: flip horizontally and vertically
frame	The frame attribute
window	The window attribute
srcWindow	The attribute of the content window
cropWindow	The attribute of the cropping window

5. Window (window)

Name	Description
x	x, relative value (0 -1000000), the same below
y	y
width	Width
height	Height
type	(Optional) Aspect ratio 0: free ratio 1: original ratio 2: 4:3

3: 16:9
4: 1:1

get-source-thumbnail

Use the interface to get the source thumbnail.

Request Mode

```
POST http://ip/mwapi/get-source-thumbnail
```

Parameter	Description
sourceId	The source ID
mediaProgress	The media progress (0 - 1000000)

Response Body

JSON structure is as follows:

```
{
  "status": 0,
  "thumbnail": {...}
}
```

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

2. Content of thumbnail

```
"thumbnail": {
  "encoding": "base64",
  "data": "..."}
}
```

Name	Description
encoding	The encoding mode
data	The encoded data

HTTP Access

[http://\[IP\]/source/\[sourceId\]](http://[IP]/source/[sourceId])

get-video-player-status

Use the interface to get the video player status information of the current presentation scene.

Request Mode

```
POST http://ip/mwapi/get-video-player-status
```

Response Body

JSON structure is as follows:

```
{
  "status":0,
  "info":[
    {
      "layerId":1,
      "playStatus":1,
      "progress":16664,
      "sourceDuration":120000,
      "sourceId":101
    }
  ]
}
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.
layerId	The layer ID, that is, the index in the layer list
sourceId	The source ID
playStatus	The play status 0: pause 1: playing
progress	The play progress (0 - 1000000)
sourceDuration	The length of the video, in ms

rename-scene

Use the interface to rename the scene.

Request Mode

POST http://ip/mwapi/rename-scene

Parameter	Description
sceneId	The ID of the editing scene
name	The new name
showId	The presentation ID

```
{  
  "sceneId":100,  
  "name":"New scene 1",  
  "showId":7  
}
```

Response Body

JSON structure is as follows:

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

1. Response Status

```
{  
  "message": "Scene is not in current presentation",  
  "status": 48  
}
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

seek-video-player

Use the interface to set the video play progress of the current scene.

Request Mode

```
POST http://ip/mwapi/seek-video-player
```

Parameter	Description
layerId	The layer ID
progress	The play progress (0 - 1000000)

Response Body

JSON structure is as follows:

```
{  
  "status": 0  
}
```

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

set-switch-action

Use the interface to simulate key events of the swap button.

Request Mode

```
POST http://ip/mwapi/set-switch-action
```

```
{  
  "action": 0,  
}
```

Parameter	Description
action	0~4, corresponding to the buttons on the devices

Response Body

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

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.
action	Key events (0 - 4)

set-video-player-status

Use the interface to set the video play status of the current scene.

Request Mode

```
POST http://ip/mwapi/set-video-player-status
```

Parameter	Description
layerId	The layer ID
playStatus	The play status 0: pause 1: playing

Response Body

JSON structure is as follows:

```
{  
  "status": 0  
}
```

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

switch-scene

Use the interface to switch the scene.

Request Mode

```
POST http://ip/mwapi/switch-scene
```

Parameter	Description
sceneId	The scene ID

Response Body

JSON structure is as follows:

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

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

2. The device is in the scene editing status, so it is not allowed to switch scenes.

```
{  
  "message": "device busy",  
  "status": 11  
}
```

upload-source-file

Use the interface to upload source files which can be pictures, video or music.

Request Mode

```
POST http://ip/mwapi/upload-source-file
```

Response Body

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

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

get-app-settings

Use the interface to obtain APP related settings.

Request Mode

```
POST http://ip/mwapi/get-app-settings
```

Response Body

```
{  
  "status": 0,  
  "need-password": 0,  
  "have-password": 0,  
  "ssdp-port": 1900,  
  "control-port": 9000,  
  "stream-port": 9110,  
  "ftp-port": 21,  
  "resolution": "1280x720",  
  "duration": 333333,  
  "video-bitrate": 2048  
}
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.
need-password	Whether a password is needed. 0: No 1: Yes
have-password	Whether the password is null. 0: Null 1: Not null
resolution	The resolution
duration	The frame rate
video-bitrate	The video bitrate
stream-port	The SRT streaming port

get-encoder-params

Use the interface to obtain the collection of encoder parameters.

Request Mode

POST http://ip/mwapi/get-encoder-params

Response Body

```
{
  "status": 0,
  "resolutions": [{...}],
  "durations": [{...}],
  "video-bitrates": [{...}],
}
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

1. Resolutions

```
[
  {
    "name": "1080P",
    "value": "1920x1080"
  },
  {
    "name": "720P",
    "value": "1080x720"
  }
]
```

Name	Description
name	The displayed name
value	Value: W x H

2. Durations

```
[
  {
    "name": "60 FPS",
    "value": "166667"
  },
  {
    "name": "59.94 FPS",
    "value": "166833"
  },
  {
    "name": "50 FPS",
    "value": "200000"
  },
  {
    "name": "30 FPS",
    "value": "333333"
  },
  {
    "name": "29.97 FPS",
    "value": "333667"
  },
  {
    "name": "25 FPS",

```

```

    "value":"400000"
  },
  {
    "name":"24 FPS",
    "value":"416667"
  },
  {
    "name":"15 FPS",
    "value":"666667"
  },
  {
    "name":"10 FPS",
    "value":"1000000"
  },
  {
    "name":"5 FPS",
    "value":"2000000"
  }
]

```

Name	Description
name	The displayed name
value	The time interval between two frames

3.Video-bitrates

```

[
  {
    "name":"8 Mbps",
    "value":"8192"
  },
  {
    "name":"6 Mbps",
    "value":"6144"
  },
  {
    "name":"4 Mbps",
    "value":"4096"
  },
  {
    "name":"2 Mbps",
    "value":"2048"
  },
  {
    "name":"1 Mbps",
    "value":"1024"
  },
  {
    "name":"768 Kbps",
    "value":"768"
  },
  {
    "name":"512 Kbps",
    "value":"512"
  },
  {
    "name":"256 Kbps",
    "value":"256"
  }
]

```

Name	Description
name	The displayed name
value	The video bitrate

set-app-password

Use the interface to set the password of the APP.

Request Mode

POST http://ip/mwapi/set-app-password

Parameter	Description
method	set-app-password
need-password	Whether a password is needed. 0: No 1: Yes
new-password	The new password, which must be filled in when "Need Password" is set.

```
{  
  "new-password": "8d969eef6ecad3c29a3a629280e686cf0c3f5d5a86aff3ca12020c923adc6c92",  
  "need-password": 1  
}
```

Response Body

```
{  
  "message": "success",  
  "status": 0  
}
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

set-encoder-format

Use the interface to modify the encoder parameter.

Request Mode

POST http://ip/mwapi/set-encoder-format

Parameter	Description
method	set-encoder-format
resolution	The resolution
duration	The frame rate
video-bitrate	The video bitrate

```
{  
  "resolution": "1920x1080",  
  "duration": 333333,  
  "video-bitrate": 2048  
}
```

Response Body

```
{  
  "message": "success",  
  "status": 0  
}
```

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

set-network-port

Use the interface to modify the network protocol port.

Request Mode

```
POST http://ip/mwapi/set-network-port
```

Parameter	Description
control-port	The control command network protocol port
stream-port	The SRT streaming network protocol port

```
{  
  "control-port":9000,  
  "stream-port":9110  
}
```

Response Body

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

Name	Description
status	0 indicates a successful data acquisition. Refer to API Status Codes to find specific description for other values.

/system/device-info

Use the interface to obtain the device information. Please check whether each sub-item of capability is true, and only when it is true, the corresponding API can be accessed.

Request Mode

```
POST /api/system/device-info
```

Response Body

```
{
  "device-name": "USB Fusion HDMI",
  "product-id": "0x506",
  "product-name": "USB Fusion HDMI",
  "hardware-rev": "A",
  "serial-number": "A506210323002",
  "firmware-ver": "1.1.202",
  "firmware-name": "Development",
  "build-time": "2021-12-17 01:07:22",
  "capability": {
    "support-timezone": true,
    "support-ntp": true,
    "support-4g": false,
    "support-station": true,
    "support-ap": true,
    "support-online-upgrade": true,
    "support-sc-control": true,
    "support-ipv6": false
  },
  "status": 0
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
product-id	The device's id
product-name	The device's name
hardware-rev	The hardware version
serial-number	The device's serial number
firmware-ver	The device's firmware version
firmware-name	The device's firmware name
build-time	The device's firmware build time
capability.support-timezone	The supported timezone
capability.support-ntp	The device supports NTP.
capability.support-4g	The device supports 4G modules.
capability.support-station	WIFI supports STA mode.
capability.support-ap	WIFI supports AP mode.
capability.support-online-upgrade	The device supports online upgrade.
capability.support-sc-control	The device supports cloud management.
capability.support-ipv6	The device supports IPv6.

/system/info

Use the interface to obtain CPU and memory information.

Request Mode

POST /api/system/info

Response Body

```
{
  "device-name": "USB Fusion HDMI",
  "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"
  },
  "status": 0
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
device-name	The device's name
uptime	The uptime, in seconds
cpu.total	The total time of CPU
cpu.idle	The idle time of CPU
cpu.usage	The CPU usage x 100
mem.total	The system's total memory, in KB
mem.avail	The system's available memory, in KB
datetime.cur-time	The system time Time format: yyyy-MM-dd HH:mm:ss
datetime.zonename	The timezone name
datetime.ntp-enable	Enables NTP.
datetime.ntp-server1	The NTP server 1
datetime.ntp-server2	The NTP server 2

/system/set-device-name

Use the interface to set the device name.

Request Mode

POST /api/system/set-device-name

Parameter	Description
name	The device name

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/system/timezone-set

Use the interface to set timezone.

Request Mode

```
POST /api/system/timezone-set
```

Parameter	Description
zonename	The timezone name

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/system/set-date-time

Use the interface to set the NTP function.

Request Mode

```
POST /api/system/set-date-time
```

Parameter	Description
ntp-enable	Whether to enable NTP
ntp-server1	The NTP server 1
ntp-server2	The NTP server 2
time	Local time Time format: yyyy-MM-dd HH:mm:ss

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/user/login

Use the interface to log in. After the user logs in successfully, the Session ID is stored in the Cookie (Cookie: sid=t2i704wbvoy51y408p588bpji010ibp0).

Request Mode

```
POST /api/user/login
```

Parameter	Description
username	The username
password	The password which is encrypted with SHA256

Response Body

```
{
  "status": 0,
  "sid": "t2i704wbvoy51y408p588bpji010ibp0"
}
```

Name	Description
status	0 indicates that the request was accepted successfully. 36 indicates that the user name or password is incorrect. Refer to API Status Codes to find specific description for other values.

Interface Example

```
// login (username: Admin, password=Admin)
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"}'
```


/user/logout

Use the interface to log out and return to the login screen.

Request Mode

```
POST /api/user/logout
```

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/user/get-all

Use the interface to obtain the user list of the system, and only the administrator has the rights.

Request Mode

POST /api/user/get-all

Response Body

```
{
  "users": [
    {
      "username": "Admin",
      "group": "Admin"
    }
  ],
  "status": 0
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
users	The user group arrays Username: user name Group: user group

/user/add

Use the interface to add a user, and only the administrator has the rights.

Request Mode

POST /api/user/add

Parameter	Description
username	The username
password	The password which is encrypted with SHA256

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/user/del

Use the interface to delete a user, and only the administrator has the rights.

Request Mode

```
POST /api/user/del
```

Parameter	Description
username	The user login name

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/user/ch-password

Use the interface to change the user's login password. The current password must be input when changing the password.

Request Mode

```
POST /api/user/ch-password
```

Parameter	Description
password	The current password which is encrypted with SHA256
new-password	The new password which is encrypted with SHA256

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/user/set-password

Use the interface to reset the password, and it does not need to input the current password. Only the administrator has the rights.

Request Mode

```
POST /api/user/set-password
```

Parameter	Description
username	The user login name
password	The new password which is encrypted with SHA256

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/network/if-info

Use the interface to obtain network card information.

Request Mode

```
POST /api/network/if-info
```

Response Body

```
{
  "device-name": "USB Fusion yxy1",
  "net": [
    {
      "enable": true,
      "iface": "eth0",
      "type": 0,
      "use-dhcp": true,
      "ipaddr": "10.10.12.166",
      "netmask": "255.255.240.0",
      "gateway": "10.10.0.1",
      "mac": "84:85:86:87:88:2e",
      "link-speed": 1000,
      "link-state": 2,
      "tx-speed-kbps": 0,
      "rx-speed-kbps": 107
    },
    {
      "enable": true,
      "iface": "wlan0",
      "type": 1,
      "mode": 1,
      "ssid": "USB-Fusion_yx_5G",
      "use-dhcp": true,
      "ipaddr": "192.168.67.1",
      "netmask": "255.255.255.0",
      "gateway": "",
      "mac": "10:2c:6b:fd:9b:78",
      "link-speed": -1,
      "link-state": 2,
      "tx-speed-kbps": 3,
      "rx-speed-kbps": 0
    },
    {
      "enable": true,
      "iface": "usb0",
      "type": 3,
      "use-dhcp": true,
      "ipaddr": "192.168.66.1",
      "netmask": "255.255.255.0",
      "gateway": "192.168.66.1",
      "mac": "8e:40:df:be:7c:fa",
      "link-speed": 480,
      "link-state": 2,
      "tx-speed-kbps": 0,
      "rx-speed-kbps": 0
    }
  ],
  "status": 0
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
device-name	The device name
net[i].enable	Whether the network card service is enabled

net[i].iface	The network card name
net[i].type	The network card type 0: Ethernet 1: WiFi 2: 4G module 3: USB
net[i].mode	The working mode of WiFi When net[i].type == 1 exists, 0: STA mode 1: AP mode
net[i].ssid	The WIFI ssid
net[i].reboot-require	WiFi reboots and takes effect.
net[i].use-dhcp	True: use DHCP to get the IP False: use the static network configuration
net[i].ipaddr	The IP address
net[i].netmask	The subnet mask
net[i].ipv6addr	The IPv6 address
net[i].gateway	The gateway address
net[i].mac	The MAC address
net[i].link-speed	The link speed 10: 10Mbps, 100: 100Mbps, 1000: 1Gbps, 2500: 2.5Gbps, 10000: 10Gbps The speed supported by USB 12: full-speed, 480: high-speed, 5000: super-speed-5g, 10000: super-speed-10g
net[i].link-state	The link state 0: down 1: disconnected 2: connected
net[i].vendor	The vendor of the 4G module
net[i].product	The product information of the 4G module
net[i].tx-speed-kbps	The sending speed (Kbps)
net[i].rx-speed-kbps	The receiving speed (Kbps)

/network/if-set

Use the interface to configure the network card.

Request Mode

```
POST /api/network/if-set
```

Parameter	Description
iface	The network card name
use-dhcp	True: use DHCP to get the IP False: Use the static network configuration
ipaddr	The IP address, which must be filled in when use-dhcp == false
netmask	The subnet mask, which must be filled in when use-dhcp == false
gateway	The gateway address, which must be filled in when use-dhcp == false

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/network/if-route

Use the interface to obtain the default route.

Request Mode

```
POST /api/network/if-route
```

Response Body

```
{  
  "iface": "",  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
ifname	The network card that the default route goes through If iface is null, it indicates that there is no route.

/network/get-dns

Use the interface to get the DNS.

Request Mode

```
POST /api/network/get-dns
```

Response Body

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

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
is-manual	Whether to set the DNS manually
dns1	DNS Null character indicates that it is not set.
dns2	DNS Null character indicates that it is not set.

/network/set-dns

Use the interface to set DNS.

Request Mode

POST /api/network/set-dns

Parameter	Description
is-manual	Whether to set DNS manually
dns1	DNS Null character indicates that it is not set.
dns2	DNS Null character indicates that it is not set.

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/network/usb-config

Use the interface to configure the USB network card

Request Mode

```
POST /api/network/usb-config
```

Parameter	Description
iface	The network card name
ipaddr	The IP address

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/network/wireless-mode

Use the interface to switch WiFi working mode.

Request Mode

```
POST /api/network/wireless-mode
```

Parameter	Description
iface	The network card name
mode	The WiFi working mode 0: STA mode 1: AP mode

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/network/wireless-state

Use the interface to check the working status of the AP.

Request Mode

```
POST /api/network/wireless-state
```

Parameter	Description
iface	The network card name

Response Body

```
{  
  "enable": true,  
  "mode": 1,  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
mode	The WiFi working mode 0: STA mode 1: AP mode
enable	Whether the network card service is enabled

/network/ap-passwd

Use the interface to set the password of the AP.

Request Mode

```
POST /api/network/ap-passwd
```

Parameter	Description
iface	The network card name
password	The password of the AP

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/network/ap-config

Use the interface to set AP.

Request Mode

POST /api/network/ap-config

Parameter	Description
iface	The network card name
ssid	The AP name
hw_mode	The working frequency of WiFi a: 5GHz g: 2.4GHz
channel	The default AP channel
country-code	Refer to ISO/IEC 3166-1
ipaddr	The IP address
dhcp-ip-start	The starting IP address allocated by DHCP
dhcp-ip-end	The ending IP address allocated by DHCP
enable-surfing	Enable surfing

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/network/ap-get-config

Use the interface to obtain AP information.

Request Mode

POST /api/network/ap-get-config

Parameter	Description
iface	The network card name

Response Body

```
{
  "ssid": "USB-Fusion_3002",
  "hw_mode": "g",
  "channel": 1,
  "country-code": "CN",
  "support-freq": [
    {
      "freq": "2.4G",
      "hw-mode": "g"
    },
    {
      "freq": "5G",
      "hw-mode": "a"
    }
  ],
  "support-channels": {
    "channels": [
      {
        "code": [
          "US",
          "CA"
        ],
        "2g-chans": [
          1,
          2,
          3,
          4,
          5,
          6,
          7,
          8,
          9,
          10,
          11
        ],
        "5g-chans": [
          36,
          40,
          44,
          48,
          149,
          153,
          157,
          161,
          165
        ]
      }
    ],
    {
      "code": [
        "AL",
        "AM",
        "AT",
        "AZ",
        "BA",
```

```
    "BE",
    "BG",
    "BY",
    "CH",
    "CY",
    "CZ",
    "DE",
    "DK",
    "EE",
    "EL",
    "ES",
    "FI",
    "FR",
    "GE",
    "HR",
    "HU",
    "IE",
    "IS",
    "IT",
    "LI",
    "LT",
    "LU",
    "LV",
    "MD",
    "ME",
    "MK",
    "MT",
    "NL",
    "NO",
    "PL",
    "PT",
    "RO",
    "RS",
    "RU",
    "SE",
    "SI",
    "SK",
    "TR",
    "UA",
    "UK"
  ],
  "2g-chans": [
    1,
    2,
    3,
    4,
    5,
    6,
    7,
    8,
    9,
    10,
    11,
    12,
    13
  ],
  "5g-chans": [
    36,
    40,
    44,
    48,
    149,
    153,
    157,
    161,
    165
  ]
},
{
  "code": [
    "JP"
  ],
  "2g-chans": [
```

```
        1,
        2,
        3,
        4,
        5,
        6,
        7,
        8,
        9,
        10,
        11,
        12,
        13
    ],
    "5g-chans": [
        36,
        40,
        44,
        48
    ]
},
{
    "code": [
        "CN"
    ],
    "2g-chans": [
        1,
        2,
        3,
        4,
        5,
        6,
        7,
        8,
        9,
        10,
        11,
        12,
        13
    ],
    "5g-chans": [
        36,
        40,
        44,
        48,
        149,
        153,
        157,
        161,
        165
    ]
}
],
"global-2g-chans": [
    1,
    2,
    3,
    4,
    5,
    6,
    7,
    8,
    9,
    10,
    11,
    12,
    13
],
"global-5g-chans": [
    36,
    40,
    44,
    48,
```

```

        149,
        153,
        157,
        161,
        165
    ]
},
"ipaddr": "192.168.67.1",
"dhcp-ip-start": "192.168.67.2",
"dhcp-ip-end": "192.168.67.254",
"first-use": false,
"ap-to-sta-reboot-effect": true,
"sta-to-ap-reboot-effect": false,
"enable-surfing": true,
"status": 0
}

```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
ssid	The AP name
hw_mode	The working frequency of WiFi a: 5GHz g: 2.4GHz
channel	The default AP channel
country-code	Refer to ISO/IEC 3166-1
support-freq	The supported frequency
ipaddr	The IP address
dhcp-ip-start	The starting IP address allocated by DHCP
dhcp-ip-end	The ending IP address allocated by DHCP
enable-surfing	Enable surfing
first-use	The first time to switch to the AP mode
ap-to-sta-reboot-effect	Whether to reboot when switching from AP to STA
sta-to-ap-reboot-effect	Whether to reboot when switching from STA to AP

/network/wifi-scan

Use the WiFi to enable WiFi to scan available hotspots.

Request Mode

```
POST /api/network/wifi-scan
```

Parameter	Description
iface	The network card name

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/network/wifi-scan-results

Use the interface to obtain the result of wifi-scan

Request Mode

POST /api/network/wifi-scan-results

Parameter	Description
iface	The network card name

Response Body

```
{
  "connect": {
    "ssid": "magewell",
    "state": 2
  },
  "user-items": [
    {
      "ssid": "magewell",
      "bssid": "94:a6:7e:61:75:c6",
      "frequency": 5220,
      "signal": -48,
      "encryption": "WPA2PSK",
      "is-auto": false,
      "is-used": true
    }
  ],
  "scan-items": [
    {
      "ssid": "TP-LINK_8EF4",
      "bssid": "18:f2:2c:1e:8e:f4",
      "frequency": 2412,
      "signal": -39,
      "encryption": "WPA2PSK",
      "is-auto": false,
      "is-used": false
    },
    {
      "ssid": "magewell_005_5GHz",
      "bssid": "14:91:82:2b:1f:6d",
      "frequency": 5765,
      "signal": -40,
      "encryption": "WPAPSK",
      "is-auto": false,
      "is-used": false
    },
    {
      "ssid": "TP-LINK_5G_8EF4",
      "bssid": "18:f2:2c:1e:8e:f6",
      "frequency": 5200,
      "signal": -42,
      "encryption": "WPA2PSK",
      "is-auto": false,
      "is-used": false
    },
    {
      "ssid": "magewell_001",
      "bssid": "f0:b4:29:54:8e:b5",
      "frequency": 2412,
      "signal": -43,
      "encryption": "WPAPSK",
      "is-auto": false,
      "is-used": false
    }
  ]
}
```

```
"status": 0
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
connect.ssid	The SSID of the connection
connect.state	The AP connection status 0: idle, not connected 1: connecting, please wait 2: connected 3: wrong password 4: connection failed 5: connection timed out
user-items[j].ssid	The AP name
user-items[j].bssid	The BSSID of the AP
user-items[j].frequency	The frequency of the AP
user-items[j].encryption	The encryption mode of the AP NONE: unencrypted WEP: WEP encryption mode WPAPSK: WPAPSK encryption mode WPA2PSK: WPA2PSK encryption mode
user-items[j].is-auto	Whether automatic connection is allowed
user-items[j].is-used	Whether it connects successfully
scan-items[i].ssid	The AP name
scan-items[i].bssid	The BSSID of the AP
scan-items[i].frequency	The frequency of the AP
scan-items[i].encryption	The encryption mode of the AP NONE: unencrypted WEP: WEP encryption mode WPAPSK: WPAPSK encryption mode WPA2PSK: WPA2PSK encryption mode
scan-items[i].is-auto	Whether automatic connection is allowed
scan-items[i].is-used	Whether it connects successfully

/network/wifi-connect

Use the interface to connect WiFi.

Request Mode

```
POST /api/network/wifi-connect
```

Parameter	Description
iface	The network card name
ssid	The AP name
password	The AP connection password If an empty string is filled in, it indicates connecting an unencrypted AP or user-items has other connected AP.
is-auto	Whether the network is connected automatically

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/network/wifi-connect-auto

Use the interface to set automatic network connection.

Request Mode

```
POST /api/network/wifi-connect-auto
```

Parameter	Description
iface	The network card name
ssid	The AP name
is-auto	Whether the network is connected automatically

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/network/wifi-connect-state

Use the interface to check the status of WiFi connecting AP.

Request Mode

```
POST /api/network/wifi-connect-state
```

Parameter	Description
iface	The network card name

Response Body

```
{  
  "ssid": "magewell",  
  "state": 2,  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
ssid	The SSID of the connected AP
state	The AP connection status 0: idle, not connected 1: connecting, please wait 2: connected 3: wrong password 4: connection failed 5: connection timed out

/network/wifi-disconnect

Use the interface to disconnect AP.

Request Mode

```
POST /api/network/wifi-disconnect
```

Parameter	Description
iface	The network card name
ssid	The AP name

Response Body

```
{  "status": 0}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/network/wifi-forget

Use the interface to remove network configuration.

Request Mode

POST /api/network/wifi-forget

Parameter	Description
iface	The network card name
ssid	The AP name

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

upgrade/online-check

Use the interface to enable online upgrade check.

Request Mode

```
POST /api/upgrade/online-check
```

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

upgrade/online-check-result

Use the interface to obtain online check results.

Request Mode

```
POST /api/upgrade/online-check-result
```

Response Body

```
{  
  "up-to-date": true,  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
up-to-date	True indicates the current firmware is up to date, otherwise it is false.
version	The latest version
size	The size of the latest version
md5	The MD5 value of the latest version
changeLog	The upgrade content of the latest version

/upgrade/upload-fw

Use the interface to upload firmware. The upload file format should be .mwf, and you should use POST multipart/form-data to upload files.

Request Mode

```
POST /upgrade/upload-fw
```

Response Body

```
{  
  "status": 0,  
  "up-to-date": true,  
  "version": "1.1.72"  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
up-to-date	Whether the firmware is the latest version
version	The firmware version to upload

/upgrade/update

Use the interface to update firmware. During the update process you can use the [/upgrade/state](#) interface to retrieve the current status.

Request Mode

```
POST /api/upgrade/update
```

Parameter	Description
is-online	False: offline upgrade True: online upgrade
mode	The upgrade mode 0: Auto, which automatically selects Upgrade/Factory/FactoryClear mode 1: Upgrade 2: Factory 3: FactoryClear
timeout	Upgrade fails with timeout (upgrade progress keeps unchanged), in seconds

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/upgrade/state

Use the interface to obtain the current firmware version and upgrade status, and only the administrator has the rights.

Request Mode

```
POST /api/upgrade/state
```

Response Body

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

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
state	The task execution status 0: idle 1: initialize and upgrade 2: upgrading 3: upgraded 4: online firmware downloading
cur-ver	The current firmware version
update-version	The latest firmware version
step	The current step number, only available when state == 2
num-steps	The total number of steps for update, only available when state == 2
step-name	The name of the current step, only available when state == 2
step-progress	The progress of the current step, only available when state ==2 Value range: 0 - 100, Unit: %
download-percent	The percentage of online download

/upgrade/clear

Use the interface to clear the upgrade status.

Request Mode

```
POST /upgrade/clear
```

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/log/clear

Use the interface to clear all the system logs, and only the administrator has the rights.

Request Mode

```
POST /api/log/clear
```

Response Body

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

/log/filter

Use the interface to filter logs.

Request Mode

POST /api/log/filter

Parameter	Description
method	get-logs
types	Log types, including all, info, warn and error, which can be separated by commas if multiple types are requested.
key	The key word for filtering, which can be empty string

Response Body

```
{
  "status": 0,
  "logs": [
    {
      "time": "2021/12/20 13:18:27.210",
      "type": "info",
      "message": "User 'Admin' (192.168.66.2) logged in"
    },
    {
      "time": "2021/12/20 11:07:29.548",
      "type": "info",
      "message": "Network changed : wlan0(10.20.0.68) connected"
    },
    ...
  ]
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
logs[i].time	The log time
logs[i].type	The log type, including info, warn and error
logs[i].context	The log content

/log/export

Use the interface to export the current system log of the device as a .html file, and only the administrator has the rights.

Request Mode

```
POST /api//log/export
```

Parameter	Description
filename	The exported filename

Request Result

The log is downloaded as a .html file and saved to a local folder.

MSG_WEB_SORKET_EVENT_CHANGE_INFO

Use WebSocket to establish a connection to monitor the device status change information.

Request Mode

```
ws://xx.xx.xx.xx/ws_up_state
```

Response Body

```
{  
  "event":4001,  
  "info":{  
    "statusType":2003  
  }  
}
```

Name	Description
event	The event type
info	The event information

event type

Name	Value	Description
MSG_WEB_SORKET_EVENT_START	4000	The start identifier, no meaning
MSG_WEB_SORKET_EVENT_CHANGE_INFO	4001	The device information change event
MSG_WEB_SORKET_EVENT_END	4002	The end identifier, no meaning

statusType

Value	Description
2000	The presentation list is changed.
2001	The presentation is switched.
2002	The scene list is changed.
2003	The current preview scene is switched.
2004	The BGM list is changed.
2005	The BGM playing status is changed.
2006	The source list is changed.
2007	The album list is changed.
2008	The audio setting is changed.
2009	The FTB enabling status is changed, and the status data can be obtained by /mwapi/get-device-status.
2010	The recording status is changed.
2011	The scene edit mode is changed, such as: {"event":4001,"info":{"editSceneld":104,"editing":1,"statusType":2011}}
2012	The source is changed (the capture source signal is plugged in or out), such as: {"event":4001,"info":{"sourceld":1,"statusType":2012}}
2013	The landing scene list is changed.
2014	The current selected landing scene is changed.
2015	The general settings of the device are changed.

get-album-files-list

Use the interface to get the album file list of the server (recordings or screenshots).

Request Mode

POST http://ip/mwapi/get-album-files-list

Parameter	Description
type	The source type 0: all 1: screenshots 2: video recordings

Response Body

JSON structure is as follows:

```
{
  "info": {
    "items": [
      {
        "createTime": 1640055676,
        "duration": 0,
        "filePath": "album/2a83adf5-3e0d-4ad0-9af5-81d8fc0fea18.jpeg",
        "id": 2,
        "md5": "md5",
        "name": "2021.12.21 11:01AM",
        "size": 502784,
        "thumbPath": "album/2a83adf5-3e0d-4ad0-9af5-81d8fc0fea18.thumb.jpeg",
        "type": 1
      },
      {
        "createTime": 1640055673,
        "duration": 1849,
        "filePath": "album/16241d75-12ea-4344-bf7f-bf683e9b850e.mp4",
        "id": 1,
        "md5": "md5",
        "name": "2021.12.21 11:01AM",
        "size": 1487872,
        "thumbPath": "album/16241d75-12ea-4344-bf7f-bf683e9b850e.thumb.jpeg",
        "type": 2
      }
    ],
    "pageIndex": 0,
    "totalCount": 2
  },
  "message": "success",
  "result": 0
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
id	The file ID
type	The file type 1: screenshots 2: video recordings
name	The source name
md5	The MD5 value of the file
filePath	The relative file path, which can be accessed through http (http://deviceIP/relative-path)
thumbPath	The relative thumbnail patch, which can be accessed through http (http://deviceIP/relative-path)
createTime	The file creation time with Unix timestamp
duration	The file duration, only valid for video, in ms

size	The file size, in Byte
------	------------------------

remove-album-file

Use the interface to remove album files.

Request Mode

```
POST http://ip/mwapi/remove-album-file
```

Parameter	Description
ids	The file ID, which should be separated with comma when there are multiple ones

Response Body

JSON structure is as follows:

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

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

get-volume-config

Use the interface to get volume information.

Request Mode

POST http://ip/mwapi/get-volume-config

Response Body

```
{
  "status": 0,
  "output": [
    {
      "id": 1,
      "db": 0,
      "mute": false
    }
  ],
  "input": [
    {
      "id": 11,
      "db": 0,
      "mute": false
    },
    {
      "id": 14,
      "db": 0,
      "mute": false
    },
    {
      "id": 17,
      "db": 0,
      "mute": false
    },
    {
      "id": 18,
      "db": 0,
      "mute": false
    },
    {
      "id": 12,
      "db": 0,
      "mute": false
    },
    {
      "id": 13,
      "db": 0,
      "mute": false
    },
    {
      "id": 19,
      "db": 0,
      "mute": false
    }
  ]
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
id	1: Output; 11: Mic; 12: HDMI 1; 13: HDMI 2; 14: BGM; 15: Video in layer 1; 16: Video in layer 2; 17: USB-C; 18: Line in; 19: USB Camera
output	The output volume list
input	The input volume list

set-volume-config

Use the interface to set volume.

Request Mode

```
POST http://ip/mwapi/set-volume-config
```

Parameter	Description
id	Set the target
db	The volume level, (-100 - 0) dB
mute	Set to mute

Response Body

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

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

get-audio-card-list

Use the interface to get the audio card device list.

Request Mode

```
POST http://ip/mwapi/get-audio-card-list
```

Response Body

```
{
  "status": 0,
  "cards": [
    {
      "card": 0,
      "name": "USB Fusion Audio",
      "system": false
    },
    {
      "card": 3,
      "name": "USB Capture HDMI+",
      "system": false
    }
  ]
}
```

Name	Description
cards	The ID of the audio card
name	The name of the audio card device
system	Whether it is an internal audio card in the system or an audio card created by the peripheral device. True: internal False: peripheral

get-audio-card-mixer

Use the interface to get the attribute of the audio card.

Request Mode

POST http://ip/mwapi/get-audio-card-mixer

```
{  
  "card": 0,  
}
```

Parameter	Description
card	The ID of the audio card

Response Body

```
{  
  "status": 0,  
  "mixer": [  
    {  
      "name": "Headphone",  
      "id": 0,  
      "val": 100,  
      "flag_c": false,  
      "flag_p": true  
    },  
    {  
      "name": "Line Gain",  
      "id": 5,  
      "val": 100,  
      "flag_c": false,  
      "flag_p": true  
    },  
    {  
      "name": "Mic Gain",  
      "id": 7,  
      "val": 100,  
      "flag_c": false,  
      "flag_p": true  
    },  
    {  
      "name": "Mic",  
      "id": 14,  
      "val": 100,  
      "flag_c": true,  
      "flag_p": false  
    },  
    {  
      "name": "Line",  
      "id": 15,  
      "val": 100,  
      "flag_c": true,  
      "flag_p": false  
    }  
  ]  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
mixer	The audio card attribute list

Name	Description
------	-------------

name	The attribute name
id	The ID of the attribute
val	The value of the attribute (0 - 100)
flag_c	Whether it is the captured volume. True: Yes False: No
flag_p	Whether it is the played volume. True: Yes False: No

get-mic-gain

Use the interface to get whether Mic gain is enabled.

Request Mode

```
POST http://ip/mwapi/get-mic-gain
```

Response Body

```
{  
  "status": 0,  
  "gain":1  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
gain	0: not enabled 1: enabled

get-mic-monitor

Use the interface to get whether the Mic monitor is enabled.

Request Mode

```
POST http://ip/mwapi/get-mic-monitor
```

Response Body

```
{  
  "status": 0,  
  "monitor": true  
}
```

Name	Description
monitor	Whether to monitor the Mic. True: monitor False: not monitor

get-usb-audio-list

Use the interface to get the USB audio device list.

Request Mode

```
POST http://ip/mwapi/get-usb-audio-list
```

```
{  
  "target": 0,  
}
```

Parameter	Description
target	0: select the audio of Webcam 1: select the audio of USB Mic 2: select the audio of USB output

Response Body

```
{  
  "status": 0,  
  "selected": false,  
  "cur-dev-path": "",  
  "devices": [  
    {  
      "dev-path": "aHc6Mywx",  
      "dev-name": "HDMI [USB Capture HDMI+] #1"  
    },  
    {  
      "dev-path": "aHc6Mywy",  
      "dev-name": "HDMI [USB Capture HDMI+] #2"  
    }  
  ]  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
selected	Whether a device is selected.
cur-dev-path	The current device
devices	The device list

set-audio-card-mixer

Use the interface to set the attribute of the audio card.

Request Mode

```
POST http://ip/mwapi/set-audio-card-mixer
```

Parameter	Description
card	The ID of the audio card
id	The ID of the audio card attribute
vol	The value of the attribute (0 - 100)

Response Body

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

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

set-mic-gain

Use the interface to enable or disable Mic gain.

Request Mode

```
POST http://ip/mwapi/set-mic-gain
```

```
{  
  "gain": 0,  
}
```

Parameter	Description
gain	0: not enabled 1: enabled

Response Body

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

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

set-mic-monitor

Use the interface to set whether to enable the Mic monitor.

Request Mode

```
POST http://ip/mwapi/set-mic-monitor
```

```
{  
  "monitor": true  
}
```

Parameter	Description
monitor	Whether to monitor the Mic. True: monitor False: not monitor

Response Body

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

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

set-usb-audio

Use the interface to select the global USB Mic and USB audio output device which can be applied to the whole presentation.

Request Mode

```
POST http://ip/mwapi/set-usb-audio
```

Name	Description
target	0: select the audio of Webcam 1: select the audio of USB Mic 2: select the audio of USB output
dev-path	The ID of the device Get the device list through the get-usb-audio-list interface.

Response Body

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

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

add-launch-scene

Use the interface to add the landing scene.

Request Mode

POST http://ip/mwapi/add-launch-scene

Parameter	Description
sourceId	The source ID

```
{
  "sourceId":4
}
```

Response Body

```
{
  "info": {
    "audioList": [
      {
        "layerId": 0,
        "muted": 0,
        "volume": 1000000
      }
    ],
    "dividerLine": {
      "color": 0,
      "width": 0,
      "x1": 500000,
      "x2": 500000,
      "y1": 0,
      "y2": 1000000
    },
    "id": 25,
    "layerList": [
      {
        "cropWindow": {
          "height": 1000000,
          "type": 1,
          "width": 1000000,
          "x": 0,
          "y": 0
        },
        "flip": 0,
        "frame": {
          "color": 16777215,
          "width": 0
        },
        "given": 0,
        "givenBg": 0,
        "rotation": 360,
        "sourceDuration": 0,
        "sourceHeight": 1461,
        "sourceId": 4,
        "sourceThumbHeight": 256,
        "sourceThumbWidth": 168,
        "sourceType": 1,
        "sourceWidth": 2224,
        "srcWindow": {
          "height": 1000000,
          "width": 856262,
          "x": 71869,
          "y": 0
        },
        "videoPolicy": 1,
      }
    ]
  }
}
```



```

        "window": {
            "height": 1000000,
            "width": 1000000,
            "x": 0,
            "y": 0
        }
    },
    {
        "cropWindow": {
            "height": 1000000,
            "type": 0,
            "width": 1000000,
            "x": 0,
            "y": 0
        },
        "flip": 0,
        "frame": {
            "color": 0,
            "width": 0
        },
        "givenBg": 0,
        "rotation": 0,
        "sourceHeight": 0,
        "sourceId": 0,
        "sourceThumbHeight": 0,
        "sourceThumbWidth": 0,
        "sourceType": 0,
        "sourceWidth": 0,
        "srcWindow": {
            "height": 1000000,
            "width": 1000000,
            "x": 0,
            "y": 0
        },
        "window": {
            "height": 0,
            "width": 0,
            "x": 0,
            "y": 0
        }
    }
},
"name": "default",
"template": 2,
"type": 3
},
"message": "success",
"status": 0
}

```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
info	The scene information

get-server-launch-scenes

Use the interface to get the landing scene list.

Request Mode

```
POST http://ip/mwapi/get-server-launch-scenes
```

Response Body

```
{  
  "sceneList": [{...}],  
  "totalCount": 5  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
sceneList	The scene list, same as get-scenes-list

get-server-settings

Use the interface to obtain the general settings of the device.

Request Mode

```
POST http://ip/mwapi/get-server-settings
```

Response Body

```
{  
  "status": 0,  
  "launchScene": 20,  
  "autoSwitch": 1,  
  "usbMirror": 0,  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
launchScene	The ID of the landing scene
autoSwitch	Once an HDMI or Webcam source is connected with the device, whether it automatically switches to the source
usbMirror	Mirror the output of USB-C to display a natural scene

set-auto-switch

Use the interface to set the switch to control whether to automatically switch to the source once an HDMI, or Webcam source is connected with the device.

Request Mode

```
POST http://ip/mwapi/set-auto-switch
```

```
{  
  "autoSwitch":1  
}
```

Name	Description
autoSwitch	0: not switch 1: automatically switch to an HDMI or Webcam source when it is connected

Response Body

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

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

set-launch-scene

Use the interface to set the current landing scene.

Request Mode

```
POST http://ip/mwapi/set-launch-scene
```

```
{  
  "launchScene":21  
}
```

Name	Description
launchScene	The landing scene ID

Response Body

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

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

set-usb-mirror

Use the interface to set whether to enable mirroring the output of USB-C.

Request Mode

```
POST http://ip/mwapi/set-usb-mirror
```

```
{  
  "usbMirror":1  
}
```

Name	Description
usbMirror	0: not enable mirroring 1: enable mirroring

Response Body

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

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

export-edid

Use the interface to export the EDID of the HDMI Source input port as a .bin file.

Request Mode

```
POST http://ip/mwapi/export-edid
```

Name	Description
source-id	The input source ID 0: HDMI 1 1: HDMI 2
file-name	The name of the exported file, ending in .bin

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

get-edid-config

Use the interface to get the EDID of the HDMI Source input port.

Request Mode

POST http://ip/mwapi/get-edid-config

Parameter	Description
source-id	The input source ID 0: HDMI 1 1: HDMI 2

Response Body

```
{
  "status": 0,
  "smart-edid": true,
  "data": "AP////////wA09wEAAQAAAAEaAQOAAAB4Au6Vo1RMmSYPUFT//4AxQEVAyUBxQIGA0QDhwAEAC0gAMPJwWoCwWIoAUB10AAAEAjqAG
HE4LUBYLEUAUB10AAAEAAAA/QAPlg+HPAAAAAAAAAAAAAAAA/ABNQUdFV0VMTAogICAgAWYCA1HxV2EQHwQTBRQgISJdX19gZWZiY2QHFgMSMgl/B
xUHUD0GwFcGAF9/AWd/AINPAADiAA9uAwwAEAC4eCEQgAECaWn2F3EAXiAA+MPAeABHYAYcRwWIFgsJQBAhGMAAJ5mIVaqUQAeMEaPMwBQHxQAA
B4AAAAAAAAAAAAzW=="
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
smart-edid	Whether to use the SmartEDID True: Yes False: No
data	The EDID information, in Base64 format, which needs to be converted to hexadecimal value when displayed

set-default-edid

Use the interface to reset the EDID of the HDMI Source input port to the default value.

Request Mode

POST http://ip/mwapi/set-default-edid

Name	Description
source-id	The input source ID 0: HDMI 1 1: HDMI 2

Response Body

```
{  
  "status": 0,  
  "smart-edid": true,  
  "data": "AP////////wA09wEAAQAAAAEaAQOAAAB4Au6Vo1RMmSYPUFT//4AxQEVAYUBxQIGA0QDhwAEAC0gAMPJwWoCwWIoAUB10AAAEAjqAG  
HE4LUBYLEUAUB10AAAEAAAA/QAPlg+HPAAAAAAAAAAAAAAAA/ABNQUdFV0VMTAogICAgAWYCA1HxV2EQHwQTBRQgISJdX19gZWZiY2QHFgMSMgl/B  
xUHUD0GwFcGAF9/AWd/AINPAADiAA9uAwwAEAC4eCEQgAECaWn2F3EAXiAA+MPAeABHYAYcRwWIFgsJQBAhGMAAJ5mIVaqUQAeMEaPMwBQHxQAA  
B4AAAAAAAAAAAAzW=="  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
smart-edid	The SmartEDID Value: true/false
data	EDID information, in Base64 format, which needs to be converted to hexadecimal value when displayed

upload-edid

Use the interface to import the EDID configuration information of the HDMI Source input port. The import file format is .bin.

Request Mode

POST http://ip/mwapi/upload-edid

Name	Description
source-id	The input source ID 0: HDMI 1 1: HDMI 2

Response Body

```
{  
  "status": 0,  
  "data": "AP////////wA09wEAAQAAAAEaAQOAAAB4Au6Vo1RMmSYPUFT//4AxQEVAyUBxQIGA0QDhwAEAC0gAMPJwWoCwWIoAUB10AAAEAjqAG  
HE4LUBYLEUAUB10AAAEAAAA/QAP1g+HPAAAAAAAAAAAAAAAA/ABNQUdFV0VMTAogICAgAWYCA1HxV2EQHwQTBRQgISJdX19gZWZiY2QHFgMSMgl/B  
xUHUD0GwFcGAF9/Awd/AINPAADiAA9uAwwAEAC4eCEQgAECaWrn2F3EAXiAA+MPAeABHYAYcRwWIFgsJQBAhGMAAJ5mIVaqUQAeMEaPMwBQHxQAA  
B4AAAAAAAAAAAAzW=="  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
data	The imported EDID information, in Base64 format, which needs to be converted to hexadecimal value when displayed

get-signal-info

Use the interface to get the status information of the HDMI Source input signal.

Request Mode

```
POST http://ip/mwapi/get-signal-info/
```

Parameter	Description
source-id	The input source ID 0: HDMI 1 1: HDMI 2

Response Body

JSON structure is as follows:

```
{
  "status": 0,
  "signal-info-types": ["video-info", "audio-info", "hdmi-info", "info-frames"],
  "video-info": {...},
  "audio-info": {...},
  "hdmi-info": {...},
  "info-frames": {...}
}
```

Response Status

```
"status": 0
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

Signal Type

```
"signal-info-types": [
  "video-info",
  "audio-info",
  "hdmi-info",
  "info-frames"
]
```

Video Signal Status (video-info: {...})

```
"video-info": {
  "width": 1920,
  "height": 1080,
  "scan": "progressive",
  "field-rate": 60.00,
  "color-depth": 8,
  "color-format": "rgb",
  "aspect-ratio": "16:9",
  "sampling": "4:4:4",
  "quant-range": "full",
  "sat-range": "full",
  "frame-struct": "2d"
}
```

Name	Description
width	The total number of pixels, horizontally
height	The total number of pixels, vertically.
scan	The scan mode, including progressive, interlaced and psf

field-rate	The frame rate, including 24, 25, 29.97, 30, 48, 50, 59.94 and 60
color-depth	The color depth, including 8, 10 and 12
color-format	The color space, including rgb, bt.601, bt.709 and bt.2020
aspect-ratio	The aspect ratio, including 16:9, 4:3, etc.
sampling	The sampling mode, including 4:2:0, 4:2:2, 4:4:4 and 4:4:4:4
quant-range	The quantization range, including limited and full
sat-range	The saturation range, including limited, extended and full
frame-struct	The frame structure, including 2d, 3d-left-right, 3d-top-bottom, 3d-left-right-half and 3d-top-bottom-half

Audio Signal Status (audio-info: {...})

```
"audio-info": {
  "codec": "lpcm",
  "num-channels": 2,
  "sample-rate": 48000,
  "bit-count": 16
}
```

Name	Description
codec	The encoding type, including lpcm, ac3, aac, etc.
num-channels	The number of channels, including 1, 2, .. 16
sample-rate	The sampling rate, including 32000, 44100, etc.
bit-count	The bit rate, including 16, 20, 24, etc.

HDMI Signal Status (hdmi-info: {...})

```
"hdmi-info": {
  "mode": "hdmi",
  "hdcv": "none",
  "vic": 0,
  "it-content": false,
  "pixel-rate": "148MHz",
  "timing-htotal": 2200,
  "timing-hactive": 1920,
  "timing-h-front-porch": 88,
  "timing-h-synct-width": 44,
  "timing-h-back-porch": 148,
  "timing-vtotal": 1125,
  "timing-vactive": 1080,
  "timing-v-front-porch": 4,
  "timing-v-synct-width": 5,
  "timing-v-back-porch": 36
}
```

Name	Description
mode	The signal mode, including HDMI and DVI
vic	The video identification code
scramble	Whether to scramble to prevent signal parsing problems. If yes, it shows true; otherwise, it is false.
clock-ratio	The CPU clock ratio multiplication, including 1 and 4
hdcv	The HDCP encryption mode, including none, hdcv-1.x and hdcv-2.2
repeat-count	The repeat count, including 0, 1, 2, 4, etc.
it-content	The IT content identifier If yes, it shows true; otherwise, it is false.
timing-mode-line	The modeline Format: pclk hdisp hsyncstart hsyncend htotal vdisp vsyncstart vsyncend vtotal [flags] flags: +hsync, -hsync, +vsync, -vsync, interlace, double-scan, sog, +csync, -csync Example: 23.86 640 656 720 800 480 481 484 497 -hsync +vsync The unit of pclk is MHz, and that of other parameters is in pixels.

Information Frame (info-frames: {...})

```

"info-frames": [
  {
    "id": "AVI",
    "type": 130,
    "version": 2,
    "length": 13,
    "checksum": 96,
    "data": "ACgAIgAAADkEAACBBw=="
  },
  {
    "id": "SPD",
    "type": 132,
    "version": 1,
    "length": 25, //bytes
    "checksum": 112,
    "data": "AQAAAAAAAAAAAAAA=="
  }
]

```

Name	Description
id	The information frame name, including AVI, SPD, etc.
type	The packet type
version	The packet version
length	The packet length
checksum	The packet checksum
data	The data is encoded in Base64 and displayed in hex.

get-video-config

Use the interface to get the configuration information of the HDMI Source video.

Request Mode

POST http://ip/mwapi/get-video-config

Parameter	Description
source-id	The input source ID 0: HDMI 1 1: HDMI 2

Response Body

```
{
  "status": 0,
  "in-auto-color-fmt": true,
  "in-color-fmt": "rgb",
  "in-auto-quant-range": true,
  "in-quant-range": "full",
  "brightness": 0,
  "contrast": 100,
  "hue": 0,
  "saturation": 100,
  "deinterlace": "none",
  "out-mirror": false,
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
in-auto-color-fmt	Whether to obtain the color space of input signal automatically If yes, it shows true; otherwise, it is false.
in-color-fmt	The color space value of input signal, including rgb, bt.601, bt.709 and bt.2020
in-auto-quant-range	Whether to obtain the quantization range of input signal automatically If yes, it shows true; otherwise, it is false.
in-quant-range	The quantization range of input signal, including full and limited
brightness	The brightness (-100 - +100)
contrast	The contrast (50 - 200)
hue	The hue (-90 - 90)
saturation	The saturation (0 - 200)
deinterlace	The deinterlacing option, including none, top-field and bottom-field
out-mirror	Whether to mirror the output scene If yes, it shows true; otherwise, it is false.

set-video-config

Use the interface to modify the configuration information of the HDMI Source video.

Request Mode

POST http://ip/mwapi/set-video-config

Parameter	Description
source-id	The source ID 0: HDMI 1 1: HDMI 2

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
source-id	The input source ID 0: HDMI 1 1: HDMI 2

Interface Example

1. Set the color space of input signal

```
{  
  "in-auto-color-fmt": false,  
  "in-color-fmt": 0  
}
```

Parameter	Description
in-auto-color-fmt	Whether to obtain the color space of input signal automatically. When modifying the configuration, the value should be false.
in-color-fmt	The color space value of input signal, including rgb, bt.601, bt.709 and bt.2020

2. Set the quantization range of input signal

```
{  
  "in-auto-quant-range": false,  
  "in-quant-range": "full"  
}
```

Parameter	Description
in-auto-quant-range	Whether to obtain the quantization range of input signal automatically. When modifying the configuration, the value should be false.
in-quant-range	The quantization range of input signal, including full and limited

3. Set the color

```
{  
  "brightness": 0,  
  "contrast": 100,  
  "hue": 0,  
  "saturation": 12  
}
```

Parameter	Description
brightness	The brightness (integer, -100 - +100)
contrast	The contrast (integer, 50 - 200)
hue	The hue (integer, -90 - 90)
saturation	The saturation (integer, 0 - 200)

4. Set deinterlacing

```
{
  "deinterlace": "none"
}
```

Parameter	Description
deinterlace	The deinterlacing option, including none, top-field and bottom-field

4. Set special effects

```
http://ip/mwapi?method=set-video-config&out-mirror=false
```

Parameter	Description
out-mirror	Whether to mirror the output scene If yes, it shows true; otherwise, it is false.

get-webcam-config

Use the interface to get the configuration information of the Webcam Source video.

Request Mode

```
GET http://ip/mwapi?method=get-webcam-config
```

Parameter	Description
method	get-webcam-config

Response Body

```
{
  "status": 0,
  "device-name": "Logical HD Camera",
  "formats": [
    {
      "fourcc": "NV12",
      "resolutions": [
        {
          "width": 1920,
          "height": 1080,
          "fps": 6000
        },
        {
          "width": 1280,
          "height": 720,
          "fps": 3000
        }
      ]
    }
  ]
  "out-mirror": false,
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
device-name	The device name
formats	The format list
out-mirror	Whether to mirror the output scene If yes, it shows true; otherwise, it is false.

set-webcam-config

Use the interface to modify the configuration information of the Webcam Source video.

Request Mode

```
GET http://ip/mwapi?method=set-webcam-config&width=1920&height=1080&fps=6000&out-mirror=true
```

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
width	The captured video width
height	The captured video height
fps	The captured video frame rate (x100)
out-mirror	Whether to mirror the output scene If yes, it shows true; otherwise, it is false.

get-note-thumbnail

Use the interface to get the thumbnails of note scenes.

Request Mode

POST http://ip/mwapi/get-note-thumbnail

Parameter	Description
method	delete-scene
sceneId	The note scene ID
isThumbnail	Whether it is a thumbnail 0: thumbnail 1: original
modificationTime	The timestamp when the note is edited

```
{
  "sceneId":102,
  "isThumbnail":0,
  "modificationTime":0
}
```

Response Body

JSON structure is as follows:

```
{
  "info":{
    "isThumbnail":0,
    "sceneId":110,
    "thumbnail":{
      "data":"",
      "encoding":"base64",
      "modificationTime":8953299
    }
  },
  "status":0
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
sceneIds	The note scene ID
isThumbnail	Whether it is a thumbnail 0: thumbnail 1: original
modificationTime	The timestamp when the note is edited
encoding	The data encoding mode (Base64)
data	The encrypted data Note: When the modificationTime in the request is the latest, the data is null, to reduce network transmission.

HTTP Access Mode

Note thumbnail: [http://\[IP\]/note/\[presentation ID\]/\[note ID\]/0](http://[IP]/note/[presentation ID]/[note ID]/0)

Note preview: [http://\[IP\]/note/\[presentation ID\]/\[note ID\]/1](http://[IP]/note/[presentation ID]/[note ID]/1)

export-output-edid

Use the interface to export the EDID of the HDMI Output port as a .bin file.

Request Mode

```
POST http://ip/mwapi/export-output-edid
```

Parameter	Description
file-name	The name of the exported file, ending in .bin

Response Body

```
{  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

get-hdmi-output-info

Use the interface to get the HDMI Output information.

Request Mode

```
POST http://ip/mwapi/get-hdmi-output-info
```

Response Body

```
{  
  "status": 0,  
  "mode": 0,  
  "data": "AP////////wA09wEAAQAAAAEaAQ0AAAB4Au6Vo1RMmSYPUFT//4AxQEVAyUBxQIGA0QDhwAEAC0gAMPJwWoCwWIoAUB10AAAEAjqAG  
HE4LUBYLEUAUB10AAAEAAAA/QAPlg+HPAAAAAAAAAAAAAAAA/ABNQUdFV0VMTAogICAgAWYCA1HxV2EQHwQTBRQgISJdX19gZWZiY2QHFgMSMgl/B  
xUHUD0GwFcGAF9/AWd/AINPAADiAA9uAwwAEAC4eCEQgAECaWn2F3EAXiAA+MPAeABHYAYcRwWIFgsJQBAhGMAAJ5mIVaqUQAeMEaPMwBQHxQAA  
B4AAAAAAAAAAAAz=="  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
mode	The output mode 0: LOOPTHRU HDMI 1 1: LOOPTHRU HDMI 2 2: PGM
data	The EDID information, in Base64 format, which needs to be converted to hexadecimal value when displayed

set-hdmi-output-mode

Use the interface to set HDMI Output mode.

Request Mode

```
POST http://ip/mwapi/set-hdmi-output-mode
```

```
{  
  "mode": 0  
}
```

Parameter	Description
mode	The output mode 0: LOOPTHRU HDMI 1 1: LOOPTHRU HDMI 2 2: PGM

Response Body

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

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

get-usb-output-config

Use the interface to get the USB Output configuration.

Request Mode

```
POST http://ip/mwapi?method=get-usb-output-config
```

Response Body

```
{
  "status": 0,
  "uvc-formats": [
    {
      "fourcc": "NV12",
      "resolutions": [
        {
          "width": 1920,
          "height": 1080,
          "fps": 6000
        },
        {
          "width": 1280,
          "height": 720,
          "fps": 3000
        }
      ]
    }
  ],
  "uac-formats": [
    {
      "sample-rate": 48000,
      "channels": 2,
      "bits-count": 16
    }
  ]
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
uvc-formats	The UVC format list
uac-formats	The UAC format list

add-presentation

Use the interface to add a new presentation.

Request Mode

```
POST http://ip/mwapi/add-presentation
```

Parameter	Description
name	The presentation name

```
{  
  "name": "New Presentation 1"  
}
```

Response Body

JSON structure is as follows:

```
{  
  "message": "success",  
  "status": 0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

delete-presentation

Use the interface to delete a presentation.

Request Mode

```
POST http://ip/mwapi/delete-presentation
```

Parameter	Description
id	The presentation ID

```
{  
  "id":8  
}
```

Response Body

JSON structure is as follows:

```
{  
  "status":0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

get-presentations-list

Use the interface to get the presentation list.

Request Mode

```
POST http://ip/mwapi/get-presentations-list
```

Response Body

JSON structure is as follows:

```
{
  "status":0,
  "presentationsList":[
    {
      "id":2,
      "name":"New presentation 2",
      "isCurrent":1,
      "sceneList":[]
    },
    {
      "id":3,
      "name":"New presentation 3",
      "isCurrent":0,
      "sceneList":[]
    }
  ]
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
id	The presentation ID
name	The presentation name
isCurrent	Whether it is the current presentation 0: No 1: Yes
sceneList	The scene list of the presentation

rename-presentation

Use the interface to rename a presentation.

Request Mode

```
POST http://ip/mwapi/rename-presentation
```

Parameter	Description
id	The presentation ID
name	The presentation name

```
{  
  "id":8,  
  "name": "New Presentation 1"  
}
```

Response Body

JSON structure is as follows:

```
{  
  "status":0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

switch-presentation

Use the interface to switch presentations.

Request Mode

```
POST http://ip/mwapi/switch-presentation
```

Parameter	Description
id	The presentation ID

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

Response Body

JSON structure is as follows:

```
{  
  "status":0  
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

check-source-exist

Use the interface to check whether the source file exists.

Request Mode

POST http://ip/mwapi/check-source-exist

Parameter	Description
sourceType	The source type 1: picture 2: video 6: music
md5	The MD5 value of the source file

Response Body

JSON structure is as follows:

```
{
  "info": {
    "find": 1,
    "source": {
      "duration": 0,
      "filePath": "2009/2009.JPG",
      "height": 1080,
      "id": 2009,
      "md5": "f7e7f0cd578a44c77aed69e7c147d676",
      "name": "IMG_3010",
      "rotation": 0,
      "secondaryType": 0,
      "thumbHeight": 144,
      "thumbWidth": 256,
      "thumbnailPath": "2009/2009.thumb.jpeg",
      "type": 1,
      "width": 1920
    }
  },
  "message": "success",
  "result": 0
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
find	0: not founded 1: the specific preset source is founded
source	The source entity

check-source-used

Use the interface to check whether the source file is used.

Request Mode

POST http://ip/mwapi/check-source-used

Parameter	Description
sourceType	The source type 1: picture 2: video 6: music
id	The source ID

Response Body

JSON structure is as follows:

```
{
  "result":0,
  "message":"success",
  "info":{
    "used":1
  }
}
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
used	0: not used 1: used by one presentation 2: used by the current preview scene, undeletable

delete-source

Use the interface to delete a source.

Request Mode

```
POST http://ip/mwapi/delete-source
```

Parameter	Description
id	The source ID
sourceType	The source type

Response Body

JSON structure is as follows:

```
{  
  "message": "The resource is previewing",  
  "status": 40  
}
```

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

get-sources-list

Use the interface to get the server source list.

Request Mode

POST http://ip/mwapi/get-sources-list

Parameter	Description
type	The source type 1: picture 2: video 6: music 0: preset source (capture + picture + video)

Response Body

JSON structure is as follows:

```
{
  "info":{
    "pageIndex":0,
    "sourceList":[
      {
        "duration":0,
        "filePath":"4/4.jpeg",
        "height":1461,
        "id":4,
        "name":"default picture",
        "thumbnailPath":"4/4.thumb.jpeg",
        "type":1,
        "width":2224
      },
      {
        "duration":0,
        "filePath":"100/100.jpeg",
        "height":1080,
        "id":100,
        "name":"ubuntu",
        "thumbnailPath":"100/100.thumb.jpeg",
        "type":1,
        "width":1920
      },
      {
        "duration":120000,
        "filePath":"101/101.mp4",
        "height":1080,
        "id":101,
        "name":"1080p-red",
        "thumbnailPath":"101/101.thumb.jpeg",
        "type":2,
        "width":1920
      },
      {
        "duration":0,
        "filePath":"1018/1018.JPG",
        "height":720,
        "id":1018,
        "name":"IMG_2878",
        "thumbnailPath":"1018/1018.thumb.jpeg",
        "type":1,
        "width":1280
      },
      {
        "duration":1234,
        "filePath":"1041/1041.mp3",
        "height":0,
        "id":1041,
```



```

        "name": "IMG_2969",
        "thumbnailPath": "1041/1041.thumb.jpeg",
        "type": 6,
        "width": 0,
        "artist": "Magewell"
    }
],
    "totalCount": 11
},
    "status": 0
}

```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
id	The source ID
type	The source type 1: picture 2: video 3: HDMI 1 4: HDMI 2 5: USB CAMERA 6: music 7: note background
name	The source name
thumbnailPath	The relative path of source thumbnail, accessing through http (http://[deviceIP]/file/relativepath)
filePath	The relative path of source file, accessing through http (http://[deviceIP]/file/relativepath)
width	The source width, valid except for music
height	The source height, valid except for music
duration	The source duration, valid only for video and music
artist	The artist, valid only for music

import-source

Use the interface to import sources.

Request Mode

POST http://ip/mwapi/import-source

Parameter	Description
fileName	The filename
md5	The MD5 value of the file
sourceType	The source type 1: picture 2: video 6: music

ps: The file has been uploaded to the server through "/mwapi/upload-source-file".

Response Body

JSON structure is as follows:

```
{
  "info":{
    "duration":0,
    "filePath":"2039/2039.png",
    "height":1080,
    "id":2039,
    "name":"IMG_3162",
    "rotation":0,
    "thumbnailPath":"2039/2039.thumb.jpeg",
    "type":1,
    "width":1440
  },
  "message":"success",
  "result":0
}
```

1. Response Status

"status": 0

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.
info	The source entity

rename-source

Use the interface to rename the source.

Request Mode

```
POST http://ip/mwapi/rename-source
```

Parameter	Description
id	The source ID
name	The new name

```
{  
  "id":2039,  
  "name":"New Source Name"  
}
```

Response Body

JSON structure is as follows:

```
{  
  "message": "success",  
  "status": 0  
}
```

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.

upload-source-file

Use the interface to upload source files.

Request Mode

```
POST http://ip/mwapi/upload-source-file
```

Response Body

JSON structure is as follows:

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

1. Response Status

```
"status": 0
```

Name	Description
status	0 indicates that the request was accepted successfully. Refer to API Status Codes to find specific description for other values.