

Ultra Stream

前言

文档简介	1.1
接口约定	1.2
API 状态码	1.3
云平台 API 状态码	1.4
设备状态掩码	1.5
存储设备状态码	1.6
API 互斥对照表	1.7
设备发现协议	1.8
设备蓝牙协议	1.9
DEMO: Node.js	1.10
DEMO: C 语言	1.11

通用接口

get-info	2.1
get-status	2.2
get-settings	2.3
start-rec	2.4
stop-rec	2.5
start-live	2.6
stop-live	2.7
ping	2.8
get-signal-info	2.9
clear-rec	2.10
clear-live	2.11
get-report	2.12
export-report	2.13

通用设置

set-name	3.1
set-app-settings	3.2
set-app-passwd	3.3
set-first-over	3.4
set-rec-control	3.5
set-softap	3.6
set-date-time	3.7
set-signal-lost-ctrl	3.8
set-ssdp	3.9
set-auto-rec	3.10
set-volume	3.11
set-video-color	3.12
set-low-latency-live	3.13
set-auto-live	3.14
set-video-input-format	3.15

set-video-output-format	3.16
set-sync-offset	3.17
add-nosignal-file	3.18
del-nosignal-file	3.19
set-nosignal-file	3.20
use-nosignal-file	3.21

存储设备管理

start-format-usb	4.1
start-format-sd	4.2
clear-format-usb	4.3
clear-format-sd	4.4
start-test-usb	4.5
stop-test-usb	4.6
clear-test-usb	4.7
get-media-files	4.8
del-media-files	4.9

文件上传

send-file-add-server	5.1
send-file-del-server	5.2
send-file-enable-server	5.3
send-file-set-server	5.4
send-file-start-test-server	5.5
send-file-stop-test-server	5.6
clear-test-send-file	5.7
send-file-add	5.8
send-file-del	5.9
send-file-clear	5.10
send-file-get-status	5.11
send-file-set-is-auto	5.12
send-file-start	5.13
send-file-stop	5.14

直播服务器

add-server	6.1
enable-server	6.2
set-server	6.3
del-server	6.4
start-test-server	6.5
stop-test-server	6.6
clear-test-server	6.7

编码参数

set-stream	7.1
set-video	7.2
set-audio	7.3

设备重置

reboot	8.1
reset-all-settings	8.2
set-enable-key-reset	8.3
set-enable-key-reboot	8.4

用户登录

login	9.1
logout	9.2

用户管理

get-users	10.1
add-user	10.2
del-user	10.3
ch-password	10.4
set-password	10.5

网络设置

set-net	11.1
disconn-wifi	11.2
open-softap	11.3
close-softap	11.4

固件更新

update	12.1
upload-update-file	12.2
cancel-download	12.3
online-update-check	12.4
clear-upgrade	12.5
clear-check-update	12.6
set-enable-check-update	12.7

EDID

import-edid	13.1
export-edid	13.2
set-default-edid	13.3
get-edid-config	13.4
set-edid-config	13.5
get-loop-through-edid	13.6
export-loop-through-edid	13.7

云平台

cloud-reg-ex	14.1
cloud-unreg-ex	14.2

文档简介

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

本文档中的 API 可适用于以下产品:

- Ultra Stream HDMI
- Ultra Stream SDI

接口约定

一、概要

- 请求协议: HTTP
- 请求方式: 默认情况下, 数据请求和提交都用 GET 方式, 文件上传用 POST 方式
- 请求 URL 格式: http://IP/usapi?method=xxx¶m1=value1¶m2=value2...
- 返回数据格式: HTTP 状态为 200 时, 返回 JSON 数据, 否则为 HTTP 对应错误
- 登录认证方式: 在 Cookie 中携带 sid=xxxxxxxx

二、返回 JSON 数据格式

格式如下, JSON 对象中的 result 属性为 [API 状态码](#), 为 0 时表示数据获取或操作成功, 否则为其它状态码。

```
{
  "result": 0,
  "cur-status": 65552,
  "last-rec-status": 0,
  "cur-time": 0,
  "box-name": "Ultra Stream C301181214002",
  ...
}
```

API 状态码

```
{
  retSendWaiting          = 31,          // 文件上传状态: 等待上传
  retLivingAuthErr        = 30,          // 直播连接状态: 认证错误
  retLivingNotset         = 29,          // 未设置直播地址
  retLivingDNS            = 28,          // 直播连接状态: DNS 解析
  retInit                 = 27,          // 初始化状态
  retLivingAuthing        = 25,          // 直播连接状态: 正在授权中
  retLivingWaiting        = 24,          // 直播连接状态: 等待连接
  retLivingConnecting     = 23,          // 直播连接状态: 正在连接服务器
  retLivingConnected      = 22,          // 直播连接状态: 服务器已连接上
  retPushReboot          = 21,
  retAudioSignalChange    = 20,
  retBlueWrite            = 19,
  retBlueRead             = 18,
  retBlueShutDown        = 17,
  retDiskOn               = 16,
  retDiskOff              = 15,
  retDiskChange           = 14,
  retSnapshotOver        = 13,
  retPushReset            = 12,
  retPushLiving           = 11,
  retPushRecord           = 10,
  retSignalChange         = 9,
  retRouteChange          = 8,
  retIPChange             = 7,
  retNetChange            = 6,
  retCancel               = 5,          // 操作已取消
  retLowSpace             = 4,          // 存储空间不足
  retLowSpeed             = 3,          // 存储设备读写速度过低
  retRunning              = 2,          // 操作正在进行中
  retRepeat               = 1,          // 重复操作
  retSucceed              = 0,          // 请求成功
  errPasswd               = -1,         // 密码错误
  errOccupied             = -2,         // 设备已被占用
  errDisconnect           = -3,         // 保留
  errDevice               = -4,
  errDisk                 = -5,
  errUnconnect           = -6,
  errKey                  = -7,
  errVersion              = -8,
  errBusy                 = -9,         // 系统繁忙
  errParam                = -10,        // 请求参数有误
  errUsage                = -11,        // 保留
  errTimeout              = -12,
  errIP                   = -13,        // 保留
  errNotFound             = -14,        // 数据不存在
  errFile                 = -15,        // 文件错误
  errNoSpace              = -16,        // 没有存储空间
  errNeedAuth             = -17,        // 未登录授权
  errSystem               = -18,        // 系统错误
  errDiskSpeed            = -19,
  errEmpty                = -20,
  errNetwork              = -21,
  errEvent                = -22,
  errCodec                = -23,
  errBlue                 = -24,
  errNoUser               = -25,        // 用户不存在
  errNoPermissin         = -26,
  errSameName            = -27,        // 重名
  errString               = -28,        // 字符不合法
  errChannelsLimited      = -29,        // 最大只允许向 2 个直播服务器推流
  err8MLimited           = -30,        // 当视频编码码率大于 4 Mbps 时, 只允许向 1 个直播服务器推流
  errFacebookLimited      = -31,        // 向 Facebook 推流时, 禁止向其他直播服务器推流
  errCodecLimited        = -32,        // 选择 HEVC 编码时, 禁止进行直播推流
  err4GLimited            = -33,        // 录制时, 单个文件大小不能超过 4G
  errMWFUnsupported       = -34,        // 固件升级包与当前产品型号或硬件版本不匹配
  errNoSignal             = -35,        // 无信号
  errSDCard               = -36,
```

```
errXinYueServer      = -37,      // 文件上传状态: 服务器错误
errAliYunOSS         = -38,      // 文件上传状态: 阿里云OSS错误
errSDNoSpace         = -39,      // SD 卡没有存储空间
errSDNoPermission    = -40,      // SD 卡没有写权限
errRTSPLimited       = -41,      // 当选择 RTSP 直播时, 只允许向 1 个直播服务器推流
errRTSP8MLimited     = -42,      // 当选择 RTSP 直播时, 视频编码码率不可大于 4 Mbps
errLockLiving        = -43,      // 锁定直播, 不允许手动停止直播
errRecParam          = -52,      // 录制参数错误
errLiveParam         = -53,      // 直播参数错误
errFTPServer         = -100,     // FTP/SFTP 服务器错误
}
```


云平台 API 状态码

```
{
  errLogin      = -200,    // 未登录
  errSn         = -109,   // 非法序列号
  errParam      = -10,    // 参数错误
  errDevice     = -4,     // 云平台拒绝该类设备注册
  errPasswd    = -1,     // 邀请码错误
  retSuccess    = 0,
  retRepeat     = 1,     // 重复注册
  retRegistering = 2,    // 正在注册
  retInit       = 27,    // 未设置
  retOnline     = 35,    // 云平台在线
  retOffline    = 36,    // 云平台离线
  retDeleted    = 104,
  retWaiting    = 103,
  retRefused    = 102,
  retAccepted   = 101,
}
```

设备状态掩码

```
{
  statusFirst      = 0x01,      // 设备第一次运行
  statusRecord     = 0x02,      // 正在录制中
  statusLiving     = 0x04,      // 正在直播中
  statusStream     = 0x08,      // 保留
  statusDiskReady  = 0x10,      // U 盘就绪
  statusRTMPReady  = 0x20,      // RTMP 直播就绪
  statusSoftAP     = 0x40,      // 无线网卡正处于 AP 模式下
  statusMIC        = 0x100,     // 保留
  statusPHONE      = 0x200,     // 保留
  statusOutput     = 0x400,     // 保留
  statusDiskTest   = 0x1000,    // 正在进行 U 盘测试
  statusBlue       = 0x2000,    // 保留
  statusUpgrade    = 0x4000,    // 正在进行固件升级
  statusNetTest    = 0x8000,    // 正在进行直播服务器测试
  statusPasswd     = 0x10000,   // 设备已设置密码
  statusOccupied   = 0x20000,   // 设备锁定（最大支持两个 APP 同时连接设备）
  statusFormatDisk = 0x100000,  // 正在格式化 U 盘
  statusFormatSD   = 0x200000,  // 正在格式化 SD 卡
  statusSearchWifi = 0x400000,  // 正在搜索 Wi-Fi
  statusConnectWifi = 0x800000, // 正在连接 Wi-Fi
  statusConnectBlue = 0x1000000, // 保留
  statusCheckUpgrade = 0x2000000, // 正在检测固件信息
  statusReset      = 0x4000000, // 设备正在重置
  stausIPv6        = 0x8000000, // 保留
  statusTestLock   = 0x10000000, // 保留
  statusReboot     = 0x20000000, // 设备正在重启
  statusSendTest   = 0x40000000, // 正在进行上传服务器测试
}
```

存储设备状态码

存储设备主要有 U 盘和 SD 卡。

```
{
  NotDisk: 0x00,           // 设备未插入
  DiskValid: 0x01,        // 设备有效
  NotSpace: 0x03,         // 设备空间已满
  NotWrite: 0x04,         // 设备无写入权限
  LowSpace: 0x05,         // 设备剩余空间不足, 低于 200 MB
  DiskInvalid: 0x06       // 设备无效
}
```

API 互斥对照表

- N: 互不影响
- 互斥: 互相影响, 不可操作
- 重复命令: 当前用户页面刷新不及时导致的异常, 提醒用户刷新或自动刷新页面
- 后者生效: 修改结果与最后提交的修改请求保持一致

请求类型	USB Format	Performance Test	Living Test	Edit Streaming Server	LIVE	REC	Encoding Parameter	Firmware Update	Reset	Reboot	Other
Usb Format	重复命令	互斥	N	N	N	互斥	N	N	N	N	N
Performance Test	互斥	重复命令	N	N	N	互斥	N	N	N	N	N
Living Test	N	N	互斥	互斥	互斥	N	互斥	N	N	N	N
Edit Streaming Server	N	N	互斥	后者生效	互斥	N	N	N	N	N	N
LIVE	N	N	互斥	互斥	重复命令	N	互斥	N	N	N	N
REC	互斥	互斥	N	N	N	重复命令	互斥	N	N	N	N
Encoding Parameter	N	N	互斥	N	互斥	互斥	后者生效	N	N	N	N
Firmware Update	互斥	互斥	互斥	互斥	互斥	互斥	互斥	重复命令	互斥	互斥	互斥
Reset	互斥	互斥	互斥	互斥	互斥	互斥	互斥	互斥	重复命令	互斥	互斥
Reboot	互斥	互斥	互斥	互斥	互斥	互斥	互斥	互斥	互斥	重复命令	互斥
Other	N	N	N	N	N	N	N	N	N	N	N



设备发现协议

可以通过 **组播方式** 或 **SSDP 协议** 发现设备。

组播方式

可以获取设备的一些基本配置和状态信息，如设备名称、序列号、工作状态、网络状态等。

- 组播地址：239.255.255.250
- 端口：2538

返回数据如下：

```
{
  "version": "1.0",
  "flag": "ssip",
  "product": "Ultra Stream",
  "boxname": "[LH-SDI]-1.2.1",
  "serialnumber": "A302190719003",
  "wifiip": "192.168.48.1",
  "ethip": "192.168.1.217",
  "status": 65600
}
```

属性	说明
version	组播通信协议版本号
flag	协议标识
product	产品类型
boxname	设备名称
serialnumber	设备序列号
wifiip	无线网 IP 地址
ethip	以太网 IP 地址
status	设备状态掩码

设备蓝牙协议

可以通过蓝牙扫描发现设备，并通过蓝牙连接与盒子交互。

设备提供的 GATT 服务

1) Service: Device info(0000180a-0000-1000-8000-00805f9b34fb)

本服务主要获取设备的基本信，characteristic 特征值如下表：

UUID	说明
00002a29-0000-1000-8000-00805f9b34fb	厂商：MAGEWELL
00002a24-0000-1000-8000-00805f9b34fb	型号，例如：ULTRA STREAM HDMI
00002a25-0000-1000-8000-00805f9b34fb	设备序列号，例如：C301170101001
00002a26-0000-1000-8000-00805f9b34fb	固件版本号，例如：1.3.11

2) Service: GATT(e20a39f4-73f5-4bc4-a12f-17d1ad07a961)

本服务主要是用来跟设备交互的。如：获取设备状态、密码认证、配置Wi-Fi、打开设备AP模式、AP信息等

UUID	别名
08590f7e-db05-467e-8757-72f6faeb13d4	Buffer-0
f4be124e-4cdc-43e2-a6bb-8628e4009e6b	Buffer-1
1de1bf3d-1bbe-43aa-9a30-b2bb5ce9671d	Buffer-2
8d7bd67d-d0b3-4a5e-8d2b-2912af174a7d	Buffer-3
3a10340a-85e3-4190-86b4-b94dc505e4f8	Buffer-4

BLE广播

BLE Name: 固定为设备序列号，为 13 位字符，以一个大写字母开头，后面是 12 个数字，例如：C301170101001，序列号校验规则为：

1. 第一个为大写字母 A-Z;
2. 第 5-6 位代表生产年份，限定为 17-27 的数字;
3. 第 7-8 位代表生产月份，限定为 01-12 的数字;
4. 第 9-10 位代表生产日期，限定为 01-31 的数字。

广播数据约定

- 广播包有两种：广播包 (Advertising Data) 、 响应包 (Scan Response) 。
- 广播包中包含若干个广播数据单元，广播数据单元也称为 AD Structure。
- 广播数据单元 = 长度值Length + AD type + AD Data。
- 长度值Length只占一个字节，并且位于广播数据单元的第一个字节。

广播包	广播数据单元			总长度	说明
	Length	AD type	AD Data		
Advertising Data	0x05	0x04	4 bytes	6 Bytes	UUID
	0x16	0xFF	21 Bytes	23Bytes	厂商自定义数据
Scan Response	0x0E	0x09	13 Bytes	15Bytes	Complete Local Name

1.UUID (6 Bytes)

UUID: "ad07a961-0000-1000-8000-00805f9b34fb"

可用于指定UUID扫描设备

数据长度	值	说明
1 byte	0x05	UUID length
1 byte	0x04	UUID flag: Incomplete List of 32-bit Service Class UUIDs
4 bytes	{0x61, 0xa9, 0x07, 0xad}	UUID value

2.厂商自定义数据(23 Bytes)

AD Data(21Bytes)定义如下:

数据长度	值	说明
1 byte	0x14	Custom data length: 自定义数据长度
1 byte	0xFF	custom data flag: 自定义数据标志
2 bytes	0x2935	vender id: 常量值
2 bytes	0x0808	product id: 常量值
1 bytes	0x01	ble data layout: 常量值
2 bytes	0x0001	net protocol version, 1.0: 常量值
4 bytes	0x01010101	box status: 设备状态掩码
4 bytes	0x00000000	box eth ip: 以太网 IP 地址
4 bytes	0x00000000	box wifi ip: 无线网 IP 地址

3.Complete Local Name

完整的设备名称, 即上文中的BLE Name, 固定为13位字符的设备序列号

设备交互

客户端通过蓝牙发现服务 (GATT Service: e20a39f4-73f5-4bc4-a12f-17d1ad07a961), 再通过BLE读/写特性值与设备交互

Service: GATT(e20a39f4-73f5-4bc4-a12f-17d1ad07a961):

本服务主要是用来跟设备交互的。如: 获取设备状态、密码认证、配置Wi-Fi、打开设备AP模式、AP信息等, Service Characteristic 如下表:

UUID	别名
08590f7e-db05-467e-8757-72f6faeb13d4	Buffer-0
f4be124e-4cdc-43e2-a6bb-8628e4009e6b	Buffer-1
1de1bf3d-1bbe-43aa-9a30-b2bb5ce9671d	Buffer-2
8d7bd67d-d0b3-4a5e-8d2b-2912af174a7d	Buffer-3
3a10340a-85e3-4190-86b4-b94dc505e4f8	Buffer-4

1.获取设备状态

步骤	操作
Step 1	APP -> Box Buffer-0: 2 Bytes = 0x5335, 2 Bytes = Index
Step 2	Box -> APP Buffer-4: 2 Bytes = 0x5336, 4 Bytes = Box Status, 2 Byte = Same index

Step 1: 通过写Buffer-0特征值请求设备状态。特征值封装: 2 Bytes常量值 (0x5335) + 2 Bytes随机数

Step 2: 通过读Buffer-4特征值获取设备状态。特征值解析: 2 Bytes常量值 (0x5336) +4 Bytes设备状态 + 2 Bytes随机数

注意: Step 2中的随机数必须与Step1中的随机数一致

2.密码认证

步骤	操作
Step 1	APP -> Box Buffer-4: 2 Bytes = 0x5333, 2 Bytes = Index Buffer-0: 2 Bytes = 0x5333, 16 Bytes = MD5(password)
Step 2	Box -> APP Buffer-4: 2 Bytes = 0x5334, 4 Bytes = retSucceed (0), errPasswd(-1), 2 Byte = Same index

Step 1 中必须先写Buffer-4特征值, 再写Buffer-0特征值

3.配置 Wi-Fi

Step 1	APP -> Box
	Buffer-0: 2 Bytes = 0x5337 2 Bytes = secuID, 0 - none, 1 - WEP, 2 - WPAPSK, 3 - WPA2PSK 2 Bytes = Index 2 Bytes = 0 - only ipv4, 1 - ipv4 or ipv6, 2 - only ipv6
	Buffer-1: 18 Bytes = Wifi Name - 1
	Buffer-2: 18 Bytes = Wifi Name - 2
	Buffer-3: 18 Bytes = Wifi Passwd
	Buffer-4: 2 Bytes = 0x5337 16 Bytes = Wifi Passwd - 2
Step 2	Box -> APP
	Buffer-0: 2 Bytes = 0x5332 2 Bytes = Same Index
	Buffer-4: 2 Bytes = 0x5332 4 Bytes = retRunning(2), retSucceed (0), errPasswd(-1), errDevice(-4), errTimeout(-12), errNotFound(-14), errEmpty(-20) 4 Bytes = WifiIP if retSucceed 4 Bytes = Box Status 2 Bytes = Version Major 2 Bytes = Version Minor

Step 1: Wi-Fi名称和密码超过18个字节时，拆分成两部分写入,写入顺序: Buffer-0 Buffer-1 Buffer-2 Buffer-3 Buffer-4

Step 2: 必须先读Buffer-0特征值，读完后才能读Buffer-4特征值

4.打开设备 AP 模式

Step 1	APP -> Box
	Buffer-0: 2 Bytes = 0x5341 2 Bytes = Index
Step 2	Box -> APP
	Buffer-4: 2 Bytes = 0x5342 4 Bytes = retSucceed (0), retRunning(2), errDevice(-4) 2 Bytes = Same index

5.AP 信息

Step 1	APP -> Box
	Buffer-0: 2 Bytes = 0x5343 2 Bytes = Index
Step 2	Box -> APP
	Buffer-4: 2 Bytes = 0x5344 2 Bytes = secuID, 0 - none, 1 - WEP, 2 - WPAPSK, 3 - WPA2PSK 2 Bytes = Same index
	Buffer-2: 2 Bytes = 0x5344 16 Bytes = Wi-Fi Name
	Buffer-3: 2 Bytes = 0x5344 16 Bytes = Wi-Fi Passwd

Step 2 中读特征值的顺序为: Buffer-4 Buffer-2 Buffer-3

6.Ping

Step 1	Box -> APP
	Buffer-1:

建立Gatt连接后，周期性读Buffer-1特征，防止Gatt连接断开。

DEMO: Node.js

本文通过 Node.js 环境下的两个实例介绍 Ultra Stream API 如何调用。

DEMO 下载链接: [ultra-stream-api-demo-nodejs.zip](#)

DEMO 目录结构:

```
ultra-stream-api-demo-nodejs
|
|-- httpUtils.js    // 基于 nodejs 的 HTTP 模块封装了 get 方法和 upload 方法
|-- xxxx.mwf       // upload.js 调用 upload-update-file 接口上传的测试固件
|-- get.js         // 通过 GET 方式调用接口获取数据
|-- upload.js      // 通过 POST 方式实现文件上传
```

环境准备

- 操作系统: 支持 macOS, Linux, Windows
- 运行环境: 建议选择 LTS 版本, 最低要求 8.x

运行方式

1. 在终端控制台进入 DEMO 目录

```
cd ultra-stream-demo-nodejs
```

2. 将 get.js 和 upload.js 中的 deviceIP 替换成测试设备的 IP 地址
3. 运行 get.js

```
node get
```

4. 运行 upload.js

```
node upload
```

DEMO: C 语言

环境准备

- 操作系统: 支持 Windows、macOS、Linux

源码编译

- 开发者自己准备相应平台(Windows/macOS/Linux/...) 的"curl sdk"
- DEMO 下载链接: [ultra-stream-api-demo-c.zip](#)
- 编译 "ultra_stream_curl.c", 链接到"libcurl"
- 生成可执行性文件 "ultra_stream_curl"

运行方式

- 在终端控制台进入 bin 目录, 将固件文件拷贝到相应平台, 执行相应平台的 ultra_stream_curl

```
cd ultra-stream-api/demo/c/bin
cp ultra_stream_hdmi_rev_c_1_2_123.mwf linux
cd linux
./ultra_stream_curl <hostip:port>
```

- 输出结果

```
***** 1. login *****
login response data:
{
  "result": 0
}

***** 2. get info *****
get info response data:
{
  "result": 0,
  "mac-addr": {
    "eth": "70:b3:d5:75:d0:4c",
    "wifi": "70:b3:d5:75:d0:4d",
    "blue": "70:b3:d5:75:d0:4e"
  },
  "snapshot": "/tmp/sbox-snapshot/sbox-quarter.jpg",
  "product": {
    "sn": "C301171116015",
    "product-id": 769,
    "hardware-ver": "C",
    "firmware-id": 1,
    "firmware-ver-s": "1.2.122",
    "factory-firmware-ver-s": "1.2.116",
    "product-name": "Ultra Stream",
    "module-name": "Ultra Stream HDMI",
    "manu-name": "MAGEWELL",
    "features": 1,
    "max-lock-count": 2
  },
  "audio-range": {
    "hdmi": {
      "max": 6.00,
      "min": -100.00,
      "def": 0.00
    },
    "mic": {
      "max": 55.25,
      "min": -12.00,
      "def": 0.00
    },
    "phone": {
```

```
        "max": 6.00,  
        "min": -57.00,  
        "def": 0.00  
    }  
},  
"codec-cap": {}  
}
```

***** 3. upload firmware *****

upload firmware response data:

```
{  
    "result": 0,  
    "up-to-date": true,  
    "version": "1.2.123",  
    "size": 12494463  
}
```

get-info 接口

获取设备基本信息，主要有网卡物理地址信息、产品相关基本信息、视频和音频相关参数取值范围等。

请求方式

```
GET http://ip/usapi?method=get-info
```

参数	说明
method	方法名称: get-info

返回数据

JSON 结构如下:

```
{
  "result": 0, // 返回状态
  "mac-addr": {}, // 网卡物理地址
  "snapshot": "", // 输入视频缩略图路径
  "product": {}, // 产品相关信息
  "nosignal": {}, // 无信号图片相关限制
  "audio-range": { // 音频相关参数取值范围
    "hdm": {},
    "mic": {},
    "phone": {}
  },
  "codec-cap": { // 编码相关参数取值范围
    "resolutions": [],
    "durations": [],
    "profile": [],
    "hevc-profile": [],
    "video-kbps": [],
    "audio-kbps": [],
    "gop-sec": [],
    "video-range": [],
    "stat-sec": [],
    "video-codec": []
  },
  "rec-control": {}, // 录制配置相关参数取值范围
  "color-range": { // 视频色彩调节相关参数取值范围
    "contrast": {},
    "brightness": {},
    "saturation": {},
    "hue": {}
  },
  "video-format": { // 视频格式相关参数取值范围
    "input-color-fmt": {},
    "output-color-fmt": {},
    "quant-range": {},
    "sat-range": {}
  },
  "send-file": { // 上传服务器相关参数取值范围
    "server": {},
    "ftp-prot": {},
    "ftp-encryption": {},
    "ftp-transfer-mode": {}
  }
}
```

返回示例

```
{
  "result": 0,
  "mac-addr": {
    "eth": "70:b3:d5:75:d5:fc",
```

```

    "wifi": "70:b3:d5:75:d5:fd",
    "blue": "70:b3:d5:75:d5:fe"
  },
  "snapshot": "/tmp/sbox-snapshot/sbox-quarter.jpg",
  "product": {
    "sn": "C301181214002",
    "product-id": 769,
    "hardware-ver": "C",
    "firmware-id": 0,
    "firmware-ver-s": "1.2.122",
    "factory-firmware-ver-s": "1.2.115",
    "product-name": "Ultra Stream",
    "module-name": "Ultra Stream HDMI",
    "manu-name": "MAGEWELL",
    "features": 1,
    "max-lock-count": 2
  },
  "nosignal": {
    "max-count": 4,
    "max-width": 1920,
    "max-height": 1080,
    "max-size-kb": 512
  },
  "audio-range": {
    "hdmi": {
      "max": 6.00,
      "min": -100.00,
      "def": 0.00
    },
    "mic": {
      "max": 55.25,
      "min": -12.00,
      "def": 0.00
    },
    "phone": {
      "max": 6.00,
      "min": -57.00,
      "def": 0.00
    }
  },
  "codec-cap": {
    "resolutions": [
      {
        "w": 480,
        "h": 360
      },
      {
        "w": 640,
        "h": 360
      },
      {
        "w": 640,
        "h": 480
      },
      {
        "w": 720,
        "h": 480
      },
      {
        "w": 720,
        "h": 576
      },
      {
        "w": 768,
        "h": 576
      },
      {
        "w": 800,
        "h": 600
      },
      {
        "w": 960,
        "h": 540
      }
    ]
  }
}

```

```

    },
    {
      "w": 1024,
      "h": 768
    },
    {
      "w": 1280,
      "h": 720
    },
    {
      "w": 1280,
      "h": 800
    },
    {
      "w": 1280,
      "h": 960
    },
    {
      "w": 1280,
      "h": 1024
    },
    {
      "w": 1440,
      "h": 900
    },
    {
      "w": 1440,
      "h": 1080
    },
    {
      "w": 1600,
      "h": 1200
    },
    {
      "w": 1920,
      "h": 1080
    }
  ],
  "durations": [
    {
      "name": "15 FPS",
      "value": 666667
    },
    {
      "name": "24 FPS",
      "value": 416667
    },
    {
      "name": "25 FPS",
      "value": 400000
    },
    {
      "name": "29.97 FPS",
      "value": 333667
    },
    {
      "name": "30 FPS",
      "value": 333333
    },
    {
      "name": "50 FPS",
      "value": 200000
    },
    {
      "name": "59.94 FPS",
      "value": 166833
    },
    {
      "name": "60 FPS",
      "value": 166667
    }
  ],
  "profile": [

```

```

{
  "name": "Baseline",
  "value": 0
},
{
  "name": "Main profile",
  "value": 1
},
{
  "name": "High profile",
  "value": 2
}
],
"hevc-profile": [
  {
    "name": "Main profile",
    "value": 0
  }
],
"video-kbps": [
  {
    "name": "256 Kbps",
    "value": 256
  },
  {
    "name": "512 Kbps",
    "value": 512
  },
  {
    "name": "768 Kbps",
    "value": 768
  },
  {
    "name": "1 Mbps",
    "value": 1024
  },
  {
    "name": "1.5 Mbps",
    "value": 1536
  },
  {
    "name": "2 Mbps",
    "value": 2048
  },
  {
    "name": "3 Mbps",
    "value": 3072
  },
  {
    "name": "4 Mbps",
    "value": 4096
  },
  {
    "name": "8 Mbps",
    "value": 8192
  }
],
"audio-kbps": [
  {
    "name": " 16 Kbps",
    "value": 16
  },
  {
    "name": " 32 Kbps",
    "value": 32
  },
  {
    "name": " 48 Kbps",
    "value": 48
  },
  {
    "name": " 64 Kbps",
    "value": 64
  }
]

```

```

    },
    {
      "name": " 96 Kbps",
      "value": 96
    },
    {
      "name": "128 Kbps",
      "value": 128
    }
  ],
  "gop-sec": [
    {
      "name": " 1 sec",
      "value": 1
    },
    {
      "name": " 2 sec",
      "value": 2
    },
    {
      "name": " 5 sec",
      "value": 5
    },
    {
      "name": "10 sec",
      "value": 10
    },
    {
      "name": "30 sec",
      "value": 30
    },
    {
      "name": "60 sec",
      "value": 60
    }
  ],
  "video-range": [
    {
      "name": "Full range (0-255)",
      "value": 1
    },
    {
      "name": "Limited range (16-235)",
      "value": 0
    }
  ],
  "stat-sec": [
    {
      "name": " 1 sec",
      "value": 1
    },
    {
      "name": " 5 sec",
      "value": 5
    },
    {
      "name": "10 sec",
      "value": 10
    },
    {
      "name": "30 sec",
      "value": 30
    },
    {
      "name": "60 sec",
      "value": 60
    }
  ],
  "video-codec": [
    {
      "name": "H.264",
      "value": 0
    },
  ],

```



```

    {
      "name": "HEVC",
      "value": 1
    }
  ]
},
"rec-control": {
  "usb-option": [
    {
      "name": "Don't record",
      "value": 0
    },
    {
      "name": "Ordinary recording",
      "value": 1
    }
  ],
  "sd-option": [
    {
      "name": "Don't record",
      "value": 0
    },
    {
      "name": "Ordinary recording",
      "value": 1
    },
    {
      "name": "Loop recording",
      "value": 2
    }
  ],
  "time-unit": [
    {
      "name": "5 minutes",
      "value": 5
    },
    {
      "name": "10 minutes",
      "value": 10
    },
    {
      "name": "30 minutes",
      "value": 30
    },
    {
      "name": "40 minutes",
      "value": 40
    },
    {
      "name": "50 minutes",
      "value": 50
    },
    {
      "name": "60 minutes",
      "value": 60
    },
    {
      "name": "90 minutes",
      "value": 90
    },
    {
      "name": "120 minutes",
      "value": 120
    }
  ],
  "file-ext": [
    {
      "name": "mp4",
      "value": 0
    },
    {
      "name": "mov",
      "value": 1
    }
  ]
}

```

```

    }
  ]
},
"color-range": {
  "contrast": {
    "max": 200,
    "min": 50,
    "def": 100
  },
  "brightness": {
    "max": 100,
    "min": -100,
    "def": 0
  },
  "saturation": {
    "max": 200,
    "min": 0,
    "def": 100
  },
  "hue": {
    "max": 90,
    "min": -90,
    "def": 0
  }
},
"video-format": {
  "input-color-fmt": [
    {
      "name": "RGB",
      "value": 1
    },
    {
      "name": "YUV BT.601",
      "value": 2
    },
    {
      "name": "YUV BT.709",
      "value": 3
    },
    {
      "name": "YUV BT.2020",
      "value": 4
    }
  ],
  "output-color-fmt": [
    {
      "name": "YUV BT.601",
      "value": 2
    },
    {
      "name": "YUV BT.709",
      "value": 3
    }
  ],
  "quant-range": [
    {
      "name": "Full range (0-255)",
      "value": 1
    },
    {
      "name": "Limited range (16-235)",
      "value": 2
    }
  ],
  "sat-range": [
    {
      "name": "Full range (0-255)",
      "value": 1
    },
    {
      "name": "Limited range (16-235)",
      "value": 2
    }
  ],

```

```

    {
      "name": "Extended GAMUT range (1-254)",
      "value": 3
    }
  ]
},
"send-file": {
  "server": [
    {
      "name": "FTP Server",
      "value": 0
    },
    {
      "name": "Google Drive",
      "value": 1
    },
    {
      "name": "Dropbox",
      "value": 2
    }
  ],
  "ftp-proto": [
    {
      "name": "FTP - File Transfer Protocol",
      "value": 0
    },
    {
      "name": "SFTP - SSH File Transfer Protocol",
      "value": 1
    }
  ],
  "ftp-encryption": [
    {
      "name": "Only use plain FTP (insecure)",
      "value": 0
    },
    {
      "name": "Require explicit FTP over TLS",
      "value": 1
    },
    {
      "name": "Require implicit FTP over TLS",
      "value": 2
    }
  ],
  "ftp-transfer-mode": [
    {
      "name": "Active",
      "value": 0
    },
    {
      "name": "Passive",
      "value": 1
    }
  ]
}
}

```

get-status 接口

实时获取设备的工作状态，主要有录制状态、直播状态、固件升级状态、U 盘格式化状态等。

请求方式

```
GET http://ip/usapi?method=get-status
```

参数	说明
method	方法名称: get-status

返回数据

JSON 结构如下:

```
{
  "result": 0, // 返回状态
  "cur-status": 65552, // 设备运行状态掩码
  "last-rec-status": 0, // U 盘上次录制失败错误码
  "last-rec-status2": 0, // SD 卡上次录制失败错误码
  "cur-time": "2019-09-26 ...", // 设备当前时间
  "box-name": "", // 设备名称
  "lock-user": [], // 手机客户端连接状态
  "sysstat": {}, // 系统状态
  "rec-status": {}, // 录制状态
  "live-status": {}, // 直播状态
  "upgrade-status": {}, // 固件升级过程状态
  "format-status": {}, // 格式化状态
  "disk-test": {}, // U 盘性能测试过程状态
  "living-test": {}, // 直播推流测试过程状态
  "check-upgrade": {}, // 在线固件检测过程状态
  "send-file": {}, // 文件上传状态
  "send-file-test": {}, // 文件上传测试过程状态
  "input-signal": {}, // 输入信号信息
  "usb": {}, // U 盘信息
  "sd": {}, // SD 卡信息
  "wifi": {}, // 无线网信息, 开启 AP 模式后, 为 AP 连接信息
  "eth": {}, // 以太网信息
  "mobile": {}, // 移动网络信息
  "upgrade": {}, // 新版本固件信息
  "downgrade": {}
}
```

返回状态

```
"result": 0
```

设备运行状态掩码

```
"cur-status": 65552
```

设备当前运行的各种状态，通过不同掩码值表示，各个状态对应的掩码值请参考 [设备状态掩码](#)。采用以下计算方式来判断：

- 1、通设备状态掩码表可知，正在进行固件升级的值为： $statusUpgrade = 0x4000$
- 2、如果 $cur-status \& statusUpgrade = statusUpgrade$ ，则说明设备正处于固件升级的状态中

系统状态

```
"sysstat": {
  "cpu": 1500
  "cpu-usage": 19
  "eth-rx": 106248
  "eth-tx": 4339448
}
```

```
"mem-free": 195612
"mem-total": 524288
"mobile-rx": 0
"mobile-tx": 0
"sd-rd": 0
"sd-wr": 0
"uptime": 331255
"usb-rd": 0
"usb-wr": 0
"wlan-rx": 0
"wlan-tx": 0
}
```

输入信号状态

```
"input-signal": {
  "status": 0,
  "cx": 0,
  "cy": 0,
  "interlaced": 0,
  "frame-rate": 0.00,
  "channel-valid": 0,
  "is-lpcm": 0,
  "bits-per-sample": 0,
  "sample-rate": 0
}
```

录制状态

```
"rec-status": {
  "result": 0,
  "run-ms": 0,
  "cur-bps": 0,
  "avg-bps": 0,
  "client-id": ""
}
```

直播状态

```
"live-status": {
  "result": 0,
  "run-ms": 0,
  "cur-bps": 0,
  "avg-bps": 0,
  "net": 0,
  "result2": 0,
  "cur-bps2": 0,
  "net2": 0,
  "client-id": ""
}
```

直播推流测试过程状态

```
"living-test": {
  "upload-bps": 0,
  "percent": 0,
  "result": 27,
  "net": 0,
  "client-id": ""
}
```

在线固件检测过程状态

```
"check-upgrade": {
  "result": 0,
  "client-id": ""
}
```

新版本固件信息

```
"upgrade": {
  "ver": "",
  "date": "",
  "size-byte": 0,
  "info": []
}
```

固件升级过程状态

```
"upgrade-status": {
  "step": 0,
  "percent": 0,
  "result": 27,
  "client-id": "",
  "mode": "none"
}
```

文件上传状态

```
"send-file": {
  "total-count-ongoing": 0,
  "total-count-done": 4,
  "disk-type": 0,
  "name": "",
  "result": 31,
  "message": "",
  "left-time": 0,
  "percent": 0
}
```

文件上传测试过程状态

```
"send-file-test": {
  "result": 27,
  "client-id": "",
  "id": 0
}
```

U 盘信息

```
"usb": {
  "disk-status": 1,
  "total-size": 61638148096,
  "used-size": 3559489536,
  "free-size": 57944440832,
  "block-size": 32768,
  "usage": 6,
  "sys-path": "/dev/sda1",
  "fs-type": "vfat",
  "mount-path": "/usr/share/web-data/media/disk0",
  "fs-label": "C472-731D",
  "write-bps": 14255282,
  "read-bps": 0,
  "free-sec": 208038,
  "file-count": 47,
  "beign-time": "2019-09-24 08:04:39",
  "end-time": "2019-09-28 07:12:44",
  "total-cache-time": 0
}
```

SD 卡信息

```
"sd": {
  "disk-status": 1,
  "total-size": 31895175168,
  "used-size": 3708043264,
}
```

```
"free-size": 24997614388,
"block-size": 16384,
"usage": 12,
"sys-path": "/dev/mmcblk1p1",
"fs-type": "vfat",
"mount-path": "/usr/share/web-data/media/disk1",
"fs-label": "9C30-9CB8",
"write-bps": 0,
"read-bps": 0,
"free-sec": 89749,
"file-count": 50,
"beign-time": "2019-09-24 07:49:57",
"end-time": "2019-09-28 07:12:44",
"total-cache-time": 103062
}
```

U 盘性能测试过程状态

```
"disk-test": {
  "read-bps": 0,
  "write-bps": 0,
  "percent": 0,
  "result": 27,
  "client-id": ""
}
```

磁盘格式化过程状态

```
"format-status": {
  "percent": 0,
  "result": 27,
  "client-id": ""
}
```

无线网信息

当 `cur-status & 0x40 = 0x40` 时, 无线网卡处于 AP 模式, 以下信息为 AP 连接信息

```
"wifi": {
  "name": "MWL1",
  "level": 0,
  "ip": "192.168.8.249",
  "mask": "255.255.255.0",
  "router": "192.168.8.1",
  "dns": "192.168.8.1"
}
```

以太网信息

```
"eth": {
  "ip": "10.10.107.212",
  "mask": "255.255.0.0",
  "router": "10.10.0.1",
  "dns": "10.0.0.3"
}
```

移动网络信息

```
"mobile": {
  "ip": "",
  "mask": "",
  "router": "",
  "dns": ""
}
```


get-settings 接口

获取设备的设置信息。

请求方式

```
GET http://ip/usapi?method=get-settings
```

参数	说明
method	方法名称: get-settings

返回数据

JSON 结构如下:

```
{
  "result": 0, // 返回状态码
  "name": "Ultra...", // 设备名称
  "passwd": 1, // App 客户端是否需要密码访问
  "is-settings": 0, // App 客户端是否能够修改设备配置
  "is-ssdp": 1, // 是否开启 UPNP
  "is-startup-rec": 1, // 设备启动时, 是否自动开启录制功能
  "is-startup-live": 1, // 设备启动时, 是否自动开启直播功能
  "is-signal-lost": 0, // 信号丢失时, 是否自动停止直播和录制功能
  "is-hdmi-rec-ctrl": 0, // 是否允许数码相机或其它设备通过 HDMI 输入信号, 控制设备开始或停止录制
  "is-auto-send-file": 0, // 是否自动 SD 卡录制的文件添加到上传列表
  "is-check-update": 0, // 是否检测有没有新的固件版本
  "is-auto-rec": 0, // 是否有信号时自动启动录制
  "is-auto-live": 0, // 是否锁定直播
  "is-low-latency": 0, // 直播时是否开启低延时模式
  "enable-key-reset": 0, // 是否允许通过长按直播按键 5 秒将设备配置参数恢复至默认值
  "enable-key-reboot": 0, // 是否允许通过长按录制按键 5 秒重新启动设备
  "audio-sync-offset": 20, // 音频时延时间
  "softap": {}, // AP 配置信息
  "date-time": {}, // 时区和时间
  "rec-control": {}, // 录制配置信息
  "video-color": {}, // 视频相关信息
  "volume": {}, // 音频相关信息
  "rec-stream": 0, // 录制码流类型
  "live-stream": 0, // 直播码流类型
  "mws-stream": 1, // App 客户端预览码流类型
  "main-stream": {}, // 主码流配置信息
  "sub-stream": {}, // 次码流配置信息
  "audio": {}, // 音频配置信息
  "eth": {}, // 以太网信息
  "wifi": {}, // Wi-Fi 信息
  "stream-server": [...], // 直播服务器列表
  "send-file-cloud": [...], // 上传服务器列表
  "video-input-format": {}, // 视频输入格式
  "video-output-format": {}, // 视频输出格式
  "use-nosignal-file": 1, // 是否启用无信号图片
  "nosignal-files": [...] // 无信号图片列表
}
```

返回示例:

```
{
  "result": 0,
  "name": "Ultra Stream C3011911111100",
  "passwd": 0,
  "is-settings": 1,
  "is-ssdp": 1,
  "is-signal-lost": 0,
  "is-hdmi-rec-ctrl": 0,
  "is-auto-send-file": 0,
```

```

"is-auto-rec": 0,
"is-low-latency": 0,
"is-auto-live": 0,
"is-check-update": 1,
"audio-sync-offset": 0,
"enable-key-reset": 1,
"enable-key-reboot": 1,
"softap": {
  "is-softap": 1,
  "is-visible": 1,
  "softap-ssid": "C301181214002",
  "softap-passwd": "81214002"
},
"date-time": {
  "timezone": "UTC",
  "is-auto": 1
},
"rec-control": {
  "mime-type": 0,
  "usb-option": 1,
  "sd-option": 1,
  "root-folder": "REC_Folder",
  "file-prefix": "VID",
  "time-unit": 5,
},
"video-color": {
  "contrast": 100,
  "brightness": 0,
  "saturation": 100,
  "hue": 0
},
"volume": {
  "is-mic": 1,
  "mic-gain": 0,
  "is-spi": 1,
  "spi-gain": 0,
  "is-phone": 1,
  "phone-gain": 0
},
"rec-stream": 0,
"live-stream": 0,
"mws-stream": 1,
"main-stream": {
  "is-auto": 0,
  "codec": 0,
  "cx": 1920,
  "cy": 1080,
  "duration": 166667,
  "kbps": 4096,
  "gop": 1,
  "fourcc": 0,
  "profile": 2,
  "cbrstat": 60,
  "fullrange": 0
},
"sub-stream": {
  "cx": 1024,
  "cy": 768,
  "duration": 333333,
  "kbps": 1024,
  "gop": 1,
  "fourcc": 0,
  "profile": 2,
  "cbrstat": 60,
  "fullrange": 1
},
"audio": {
  "sample-rate": 48000,
  "channels": 2,
  "kbps": 128
},
"eth": {
  "is-dhcp": 1,

```

```

    "ip": "",
    "mask": "",
    "router": "",
    "dns": ""
  },
  "wifi": {
    "is-dhcp": 1,
    "ip": "",
    "mask": "",
    "router": "",
    "dns": ""
  },
  "stream-server": [
    {
      "id": 0,
      "type": 0,
      "url": "192.168.1.123:345/live",
      "key": "aa",
      "is-auth": 0,
      "user": "",
      "passwd": "",
      "is-use": 0,
      "token": "",
      "net-mode": 1,
      "name": "192.168.1.123"
    }
  ],
  "send-file-cloud": [
    {
      "id": 0,
      "is-use": 1,
      "type": 2,
      "account-id": "dbid:AABnr7BK6R05Plz5qH2508Eu95LI2-Lpcsc",
      "photo-path": "",
      "user-name": "qin qiu",
      "access-token": "",
      "refresh-token": "",
      "dir-name": "()",
      "net-mode": 2
    }
  ],
  "video-input-format": {
    "is-color-fmt": 0,
    "color-fmt": 1,
    "is-quant-range": 0,
    "quant-range": 1
  },
  "video-output-format": {
    "is-color-fmt": 0,
    "color-fmt": 3,
    "is-quant-range": 0,
    "quant-range": 2,
    "is-sat-range": 0,
    "sat-range": 2
  },
  "use-nosignal-file": 1,
  "nosignal-files": [
    {
      "id": 0,
      "is-use": 0,
      "is-edit": 0,
      "file-path": "/no-signal/default0.jpg",
      "time": 0
    },
    {
      "id": 1,
      "is-use": 1,
      "is-edit": 0,
      "file-path": "/no-signal/default1.jpg",
      "time": 0
    }
  ]
}

```


start-rec 接口

开启录制功能。

开启后可以调用 [stop-rec](#) 接口进行停止。

请求方式

```
GET http://ip/usapi?method=start-rec
```

参数	说明
method	方法名称: start-rec

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

stop-rec 接口

停止录制功能。

请求方式

```
GET http://ip/usapi?method=stop-rec
```

参数	说明
method	方法名称: stop-rec

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

start-live 接口

开启直播功能。

开启后可以调用 [stop-live](#) 接口进行停止。

请求方式

```
GET http://ip/usapi?method=start-live
```

参数	说明
method	方法名称: start-live

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

stop-live 接口

停止直播功能。

请求方式

```
GET http://ip/usapi?method=stop-live
```

参数	说明
method	方法名称: stop-live

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

ping 接口

判断设备是否可以访问，无需登录。

在 [固件更新](#)、[重置设备](#)、[修改 IP 地址](#) 等操作完成后，设备需要重启，可以通过该接口判断设备是否已经重启完成。

请求方式

```
GET http://ip/usapi?method=ping
```

参数	说明
method	ping

返回数据

```
{  
  "result": 0,  
  "cur-status": 65552  
}
```

属性	说明
result	0: 设备可以访问。返回其它值请参考 API 状态码 。
cur-status	设备当前工作状态掩码值。相关掩码值请参考 设备状态掩码 。

get-signal-info 接口

获取视频和音频输入信号信息。

请求方式

```
GET http://ip/usapi?method=get-signal-info
```

参数	说明
method	方法名称: get-signal-info

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

clear-rec 接口

启动录制后，会处于录制成功或失败的状态，调用该接口可以重置为初始化状态 (27)。

请求方式

```
GET http://ip/usapi?method=clear-rec
```

参数	说明
method	方法名称: clear-clear-rec

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

clear-live 接口

开启直播后，会处于直播成功或失败的状态，调用该接口可以重置为初始化状态 (27)。

请求方式

```
GET http://ip/usapi?method=clear-live
```

参数	说明
method	方法名称: clear-live

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

get-report 接口

获取设备信息、状态和配置的相关信息。

请求方式

```
GET http://ip/usapi?method=get-report
```

参数	说明
method	方法名称: get-report

返回数据

```
{  
  <div class="report-summary">  
    ...  
    <h2>DEVICE</h2>  
    ...  
    <h2>STATUS</h2>  
    ...  
    <h2>SETTINGS</h2>  
    ...  
  </div>  
}
```

export-report 接口

将设备信息、状态和配置等相关信息以 html 文件保存到本地。

请求方式

```
GET http://ip/usapi?method=export-report&file-name=...
```

参数	说明
method	方法名称: export-report
file-name	保存的文件路径

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

set-name 接口

设置设备名称。

请求方式

```
GET http://ip/usapi?method=set-name&name=xxx
```

参数	说明
method	方法名称: set-name
name	设备名称, 字符要求: 1、1 - 32 个字符 2、由 A-Z, a-z, 0-9, 空格 ._-+'[]() 组成, 并且不能以空格开头或结束

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

set-app-settings 接口

控制 App 客户端是否能够修改设备配置。

请求方式

```
GET http://ip/usapi?method=set-app-settings&is-settings=0
```

参数	说明
method	方法名称: set-app-settings
is-settings	设置权限 0: 关闭, APP 客户端无法修改设备配置 1: 开启, 允许 APP 客户端修改设备配置

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

set-app-passwd 接口

设置或修改 App 客户端和设备的配对密码。

请求方式

```
// 启用密码
GET http://ip/usapi?method=set-app-passwd&is-pass=1&new-pass=xxx

// 禁用密码, 需要输入原密码
GET http://ip/usapi?method=set-app-passwd&is-pass=0&pass=xxx

// 修改密码, 需要输入原密码
GET http://ip/usapi?method=set-app-passwd&is-pass=1&pass=xxx&new-pass=xxx
```

参数	说明
method	方法名称: set-app-passwd
is-pass	配对密码启用状态 0: 禁用 1: 启用
pass	原密码, MD5 加密, 禁用或修改时必填
new-pass	新密码, MD5 加密, 启用或修改时必填

返回数据

```
{
  "result": 0
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

set-first-over 接口

首次访问设备时，需要调用该接口进行初始化设置，主要是设置 **设备名称**、**配对密码** 和 **时区时间**。

通过 `get-status` 接口可以获取到以下信息：

```
{
  "cur-status": 65552           // 设备运行状态掩码
  ...
}
```

设备首次运行的值为：`statusFirst(0x01)`，如果 `cur-status & statusFirst = statusFirst`，那么说明设备是首次运行

请求方式

```
// 启用配对密码
GET http://ip/usapi?method=set-first-over&name=xxx&enable-passwd=1&passwd=xxx&timezone=Asia/Shanghai&is-auto=1&time=

// 禁用配对密码
GET http://ip/usapi?method=set-first-over&name=xxx&enable-passwd=0&timezone=xxx&is-auto=1&time=2019-09-10+15:00:00
```

参数	说明
method	方法名称: set-first-over
name	设备名称，字符要求： 1、1 - 32 个字符 2、由 A-Z, a-z, 0-9, 空格 . _ - + ' [] () 组成，并且不能以空格开头或结束
enable-passwd	配对密码启用状态 0: 禁用 1: 启用
passwd	密码，MD5 加密，enable-passwd = 1 时必填
time	密码，MD5 加密，enable-passwd = 1 时必填
passwd	密码，MD5 加密，enable-passwd = 1 时必填
timezone	时区，如: Asia/Shanghai，具体值请参考set-date-time.md
is-auto	是否自动获取时间，0: 手动设置，1: 根据时区自动获取
time	手动设置时间，字符串类型，格式如: 2019-09-10 15:10:00 当 is-auto=1 时，time="" 即可

返回数据

```
{
  "result": 0
}
```

属性	说明
result	返回状态，0: 操作成功，返回其它值请参考 API 状态码

set-rec-control 接口

新增于: V1.3

修改录制配置信息。

通过 [get-info](#) 可以获取到录制相关参数的取值范围:

```
"rec-control": {
  "usb-option": [],      // U 盘录制选项列表
  "sd-option": [],      // SD 卡录制选项列表
  "time-unit": [],     // 录制周期列表, 单位: 分钟
  "file-ext": []       // 视频文件后缀列表
}
```

请求方式

```
GET http://ip/usapi?method=set-rec-control&usb-option=0&sd-option=2&root-folder=xxx&file-name-mode=xxx&time-unit=5
```

参数	说明
method	方法名称: set-rec-control
usb-option	U 盘录制选项 0: 关闭录制 1: 普通录制
sd-option	SD 卡录制选项 0: 关闭录制 1: 普通录制 2: 循环录制
mime-type	录制视频格式 0: mp4 1: mov
root-folder	录制目录, 默认值为 REC_Folder, 最多支持两级目录, 不同目录用 / 分割 (如: a/b)。 字符要求: 由 A-Z, a-z, 0-9, 空格, _-+[]() 组成, 并且不能以空格或 / 开头或结束
file-name-mode	录制文件名称前缀 0: 自定义, 默认值为 VID, 字符要求: 由 A-Z, a-z, 0-9, 空格, _-+[]() 组成, 并且不能以空格开头或结束 1: 以文件创建时间命名
time-unit	录制周期, 单位: 分钟。录制时, 单个文件不得超过 4G, 即 time-unit * 录制编码率 <= 4G

返回数据

```
{
  "result": 0
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

set-softap 接口

新增于: V1.3

修改 AP 配置。

请求方式

```
GET http://ip/usapi?method=set-softap&is-softap=0&is-visible=1&softap-ssid=xxx&softap-passwd=xxx
```

参数	说明
method	方法名称: set-softap
is-softap	保留, 默认值: 1
is-visible	保留, 默认值: 1
softap-ssid	保留, 默认值: 产品序列号
softap-passwd	SSID 密码, 明文存储, 字符要求: 1、1 - 32 个字符 2、由 A-Z, a-z, 0-9, 空格 ._-+'[]() 组成, 并且不能以空格开头或结束

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

set-date-time 接口

新增于: V1.3

设置时区和时间。

请求方式

```
GET http://ip/usapi?method=set-date-time&timezone=Asia/Shanghai&is-auto=1&time=2019-09-10+15:00:00
```

参数	说明
method	方法名称: set-date-time
timezone	时区, 如: Asia/Shanghai, 具体值请参考下文常见时区
is-auto	是否自动获取时间, 0: 手动设置, 1: 根据时区自动获取
time	手动设置时间, 字符串类型, 格式如: 2019-09-10 15:10:00 当 is-auto=1 时, time="" 即可

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

常见时区

```
[  
  'Africa/Abidjan',  
  'Africa/Accra',  
  'Africa/Addis_Ababa',  
  'Africa/Algiers',  
  'Africa/Asmara',  
  'Africa/Bamako',  
  'Africa/Bangui',  
  'Africa/Banjul',  
  'Africa/Bissau',  
  'Africa/Blantyre',  
  'Africa/Brazzaville',  
  'Africa/Bujumbura',  
  'Africa/Cairo',  
  'Africa/Casablanca',  
  'Africa/Ceuta',  
  'Africa/Conakry',  
  'Africa/Dakar',  
  'Africa/Dar_es_Salaam',  
  'Africa/Djibouti',  
  'Africa/Douala',  
  'Africa/El_Aaiun',  
  'Africa/Freetown',  
  'Africa/Gaborone',  
  'Africa/Harare',  
  'Africa/Johannesburg',  
  'Africa/Juba',  
  'Africa/Kampala',  
  'Africa/Khartoum',  
  'Africa/Kigali',  
  'Africa/Kinshasa',  
  'Africa/Lagos',  
  'Africa/Libreville',  
  'Africa/Lome',  
]
```

'Africa/Luanda',
'Africa/Lubumbashi',
'Africa/Lusaka',
'Africa/Malabo',
'Africa/Maputo',
'Africa/Maseru',
'Africa/Mbabane',
'Africa/Mogadishu',
'Africa/Monrovia',
'Africa/Nairobi',
'Africa/Ndjamena',
'Africa/Niamey',
'Africa/Nouakchott',
'Africa/Ouagadougou',
'Africa/Porto-Novo',
'Africa/Sao_Tome',
'Africa/Timbuktu',
'Africa/Tripoli',
'Africa/Tunis',
'Africa/Windhoek',
'America/Adak',
'America/Anchorage',
'America/Anguilla',
'America/Antigua',
'America/Araguaina',
'America/Argentina/Buenos_Aires',
'America/Argentina/Catamarca',
'America/Argentina/ComodRivadavia',
'America/Argentina/Cordoba',
'America/Argentina/Jujuy',
'America/Argentina/La_Rioja',
'America/Argentina/Mendoza',
'America/Argentina/Rio_Gallegos',
'America/Argentina/Salta',
'America/Argentina/San_Juan',
'America/Argentina/San_Luis',
'America/Argentina/Tucuman',
'America/Argentina/Ushuaia',
'America/Aruba',
'America/Asuncion',
'America/Atikokan',
'America/Atka',
'America/Bahia',
'America/Bahia_Banderas',
'America/Barbados',
'America/Belem',
'America/Belize',
'America/Blanc-Sablon',
'America/Boa_Vista',
'America/Bogota',
'America/Boise',
'America/Buenos_Aires',
'America/Cambridge_Bay',
'America/Campo_Grande',
'America/Cancun',
'America/Caracas',
'America/Cayenne',
'America/Cayman',
'America/Chicago',
'America/Chihuahua',
'America/Costa_Rica',
'America/Creston',
'America/Cuiaba',
'America/Curacao',
'America/Danmarkshavn',
'America/Dawson',
'America/Dawson_Creek',
'America/Denver',
'America/Detroit',
'America/Dominica',
'America/Edmonton',
'America/Eirunepe',
'America/El_Salvador',

'America/Ensenada',
'America/Fort_Nelson',
'America/Fort_Wayne',
'America/Fortaleza',
'America/Glace_Bay',
'America/Godthab',
'America/Goose_Bay',
'America/Grand_Turk',
'America/Grenada',
'America/Guadeloupe',
'America/Guatemala',
'America/Guayaquil',
'America/Guyana',
'America/Halifax',
'America/Havana',
'America/Hermosillo',
'America/Indiana/Indianapolis',
'America/Indiana/Knox',
'America/Indiana/Marengo',
'America/Indiana/Petersburg',
'America/Indiana/Tell_City',
'America/Indiana/Vevay',
'America/Indiana/Vincennes',
'America/Indiana/Winamac',
'America/Indianapolis',
'America/Inuvik',
'America/Iqaluit',
'America/Jamaica',
'America/Juneau',
'America/Kentucky/Louisville',
'America/Kentucky/Monticello',
'America/Kralendijk',
'America/La_Paz',
'America/Lima',
'America/Los_Angeles',
'America/Louisville',
'America/Lower_Princes',
'America/Maceio',
'America/Managua',
'America/Manaus',
'America/Marigot',
'America/Martinique',
'America/Matamoros',
'America/Mazatlan',
'America/Mendoza',
'America/Menominee',
'America/Merida',
'America/Metlakatla',
'America/Mexico_City',
'America/Miquelon',
'America/Moncton',
'America/Monterrey',
'America/Montevideo',
'America/Montreal',
'America/Montserrat',
'America/Nassau',
'America/New_York',
'America/Nipigon',
'America/Nome',
'America/Noronha Atlantic islands',
'America/North_Dakota/Beulah',
'America/North_Dakota/Center',
'America/North_Dakota/New_Salem',
'America/Ojinaga',
'America/Panama',
'America/Pangnirtung',
'America/Paramaribo',
'America/Phoenix',
'America/Port-au-Prince',
'America/Port_of_Spain',
'America/Porto_Acre',
'America/Porto_Velho',
'America/Puerto_Rico',

'America/Punta_Arenas',
'America/Rainy_River',
'America/Rankin_Inlet',
'America/Recife',
'America/Regina',
'America/Resolute',
'America/Rio_Branco',
'America/Rosario',
'America/Santa_Isabel',
'America/Santarem',
'America/Santiago',
'America/Santo_Domingo',
'America/Sao_Paulo',
'America/Scoresbysund',
'America/Shiprock',
'America/Sitka',
'America/St_Barthelemy',
'America/St_Johns',
'America/St_Kitts',
'America/St_Lucia',
'America/St_Thomas',
'America/St_Vincent',
'America/Swift_Current',
'America/Tegucigalpa',
'America/Thule',
'America/Thunder_Bay',
'America/Tijuana',
'America/Toronto',
'America/Tortola',
'America/Vancouver',
'America/Virgin',
'America/Whitehorse',
'America/Winnipeg',
'America/Yakutat',
'America/Yellowknife',
'Asia/Aden',
'Asia/Almaty',
'Asia/Amman',
'Asia/Anadyr',
'Asia/Aqtai',
'Asia/Aqtobe',
'Asia/Ashgabat',
'Asia/Ashkhabad',
'Asia/Atyrau',
'Asia/Baghdad',
'Asia/Bahrain',
'Asia/Baku',
'Asia/Bangkok',
'Asia/Barnaul',
'Asia/Beirut',
'Asia/Bishkek',
'Asia/Brunei',
'Asia/Calcutta',
'Asia/Chita',
'Asia/Choibalsan',
'Asia/Chongqing',
'Asia/Chungking',
'Asia/Colombo',
'Asia/Dacca',
'Asia/Damascus',
'Asia/Dhaka',
'Asia/Dili',
'Asia/Dubai',
'Asia/Dushanbe',
'Asia/Famagusta',
'Asia/Gaza',
'Asia/Harbin',
'Asia/Hebron West',
'Asia/Ho_Chi_Minh',
'Asia/Hong_Kong',
'Asia/Hovd',
'Asia/Irkutsk',
'Asia/Istanbul',

'Asia/Jakarta',
'Asia/Jayapura',
'Asia/Jerusalem',
'Asia/Kabul',
'Asia/Kamchatka',
'Asia/Karachi',
'Asia/Kashgar',
'Asia/Kathmandu',
'Asia/Katmandu',
'Asia/Khandyga',
'Asia/Kolkata',
'Asia/Krasnoyarsk',
'Asia/Kuala_Lumpur',
'Asia/Kuching',
'Asia/Kuwait',
'Asia/Macao',
'Asia/Macau',
'Asia/Magadan',
'Asia/Makassar',
'Asia/Manila',
'Asia/Muscat',
'Asia/Nicosia',
'Asia/Novokuznetsk',
'Asia/Novosibirsk',
'Asia/Omsk',
'Asia/Oral',
'Asia/Phnom_Penh',
'Asia/Pontianak',
'Asia/Pyongyang',
'Asia/Qatar',
'Asia/Qyzylorda',
'Asia/Rangoon',
'Asia/Riyadh',
'Asia/Saigon',
'Asia/Sakhalin',
'Asia/Samarkand',
'Asia/Seoul',
'Asia/Shanghai',
'Asia/Singapore',
'Asia/Srednekolymsk',
'Asia/Taipei',
'Asia/Tashkent',
'Asia/Tbilisi',
'Asia/Tehran',
'Asia/Thimbu',
'Asia/Thimphu',
'Asia/Tokyo',
'Asia/Tomsk',
'Asia/Ujung_Pandang',
'Asia/Ulaanbaatar',
'Asia/Ulan_Bator',
'Asia/Urumqi',
'Asia/Ust-Nera',
'Asia/Vientiane',
'Asia/Vladivostok',
'Asia/Yangon',
'Asia/Yakutsk',
'Asia/Yekaterinburg',
'Asia/Yerevan',
'Atlantic/Azores',
'Atlantic/Bermuda',
'Atlantic/Canary',
'Atlantic/Cape_Verde',
'Atlantic/Faeroe',
'Atlantic/Faroe',
'Atlantic/Jan_Mayen',
'Atlantic/Madeira',
'Atlantic/Reykjavik',
'Atlantic/South_Georgia',
'Atlantic/St_Helena',
'Atlantic/Stanley',
'Australia/ACT',
'Australia/Adelaide',

'Australia/Brisbane',
'Australia/Broken_Hill',
'Australia/Canberra',
'Australia/Currie',
'Australia/Darwin',
'Australia/Eucla',
'Australia/Hobart',
'Australia/LHI',
'Australia/Lindeman',
'Australia/Lord_Howe',
'Australia/Melbourne',
'Australia/NSW',
'Australia/North',
'Australia/Perth',
'Australia/Queensland',
'Australia/South',
'Australia/Sydney',
'Australia/Tasmania',
'Australia/Victoria',
'Australia/West',
'Australia/Yancowinna',
'Europe/Amsterdam',
'Europe/Andorra',
'Europe/Astrakhan',
'Europe/Athens',
'Europe/Belfast',
'Europe/Belgrade',
'Europe/Berlin',
'Europe/Bratislava',
'Europe/Brussels',
'Europe/Bucharest',
'Europe/Budapest',
'Europe/Busingen',
'Europe/Chisinau',
'Europe/Copenhagen',
'Europe/Dublin',
'Europe/Gibraltar',
'Europe/Guernsey',
'Europe/Helsinki',
'Europe/Isle_of_Man',
'Europe/Istanbul',
'Europe/Jersey',
'Europe/Kaliningrad',
'Europe/Kiev',
'Europe/Kirov',
'Europe/Lisbon',
'Europe/Ljubljana',
'Europe/London',
'Europe/Luxembourg',
'Europe/Madrid',
'Europe/Malta',
'Europe/Mariehamn',
'Europe/Minsk',
'Europe/Monaco',
'Europe/Moscow',
'Europe/Nicosia',
'Europe/Oslo',
'Europe/Paris',
'Europe/Podgorica',
'Europe/Prague',
'Europe/Riga',
'Europe/Rome',
'Europe/Samara',
'Europe/San_Marino',
'Europe/Sarajevo',
'Europe/Saratov',
'Europe/Simferopol',
'Europe/Skopje',
'Europe/Sofia',
'Europe/Stockholm',
'Europe/Tallinn',
'Europe/Tirane',
'Europe/Tiraspol',

'Europe/Ulyanovsk',
'Europe/Uzhgorod',
'Europe/Vaduz',
'Europe/Vatican',
'Europe/Vienna',
'Europe/Vilnius',
'Europe/Volgograd',
'Europe/Warsaw',
'Europe/Zagreb',
'Europe/Zaporozhye',
'Europe/Zurich',
'Indian/Antananarivo',
'Indian/Chagos',
'Indian/Christmas',
'Indian/Cocos',
'Indian/Comoro',
'Indian/Kerguelen',
'Indian/Mahe',
'Indian/Maldives',
'Indian/Mauritius',
'Indian/Mayotte',
'Indian/Reunion',
'Pacific/Apia',
'Pacific/Auckland',
'Pacific/Bougainville',
'Pacific/Chatham',
'Pacific/Chuuk',
'Pacific/Easter',
'Pacific/Efate',
'Pacific/Enderbury',
'Pacific/Fakaofu',
'Pacific/Fiji',
'Pacific/Funafuti',
'Pacific/Galapagos',
'Pacific/Gambier',
'Pacific/Guadalcanal',
'Pacific/Guam',
'Pacific/Honolulu',
'Pacific/Johnston',
'Pacific/Kiritimati',
'Pacific/Kosrae',
'Pacific/Kwajalein',
'Pacific/Majuro',
'Pacific/Marquesas',
'Pacific/Midway',
'Pacific/Nauru',
'Pacific/Niue',
'Pacific/Norfolk',
'Pacific/Noumea',
'Pacific/Pago_Pago',
'Pacific/Palau',
'Pacific/Pitcairn',
'Pacific/Pohnpei',
'Pacific/Ponape',
'Pacific/Port_Moresby',
'Pacific/Rarotonga',
'Pacific/Saipan',
'Pacific/Samoa',
'Pacific/Tahiti',
'Pacific/Tarawa',
'Pacific/Tongatapu',
'Pacific/Truk',
'Pacific/Wake',
'Pacific/Wallis',
'Pacific/Yap',
'UTC'

]

set-signal-lost-ctrl 接口

新增于: V1.3

控制输入信号丢失时, 录制或直播是否自动停止。

请求方式

```
GET http://ip/usapi?method=set-signal-lost-ctrl&is-signal-lost=0
```

参数	说明
method	方法名称: set-signal-lost-ctrl
is-signal-lost	信号丢失时, 是否停止。0: 否, 1: 是

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

set-ssdp 接口

新增于: V1.3

是否开启 UPNP。

请求方式

```
GET http://ip/usapi?method=set-ssdp&is-ssdp=1
```

参数	说明
method	方法名称: set-ssdp
is-ssdp	是否启用 0: 关闭 1: 启用

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

set-auto-rec 接口

新增于: V1.3

是否允许输入信号锁定时自动启动录制，输入信号丢失时自动停止录制。

请求方式

```
GET http://ip/usapi?method=set-auto-rec&is-auto-rec=1
```

参数	说明
method	方法名称: set-auto-rec
is-auto-rec	是否允许 0: 不允许 1: 允许

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

set-volume 接口

新增于: V1.3

调节输入信号、耳机、麦克风音量的增益。

通过 [get-info](#) 可以获取到录制相关参数的取值范围:

```
"audio-range": {  
  "hdm": [], // 输入信号音量增益取值范围  
  "mic": [], // 麦克风音量增益取值范围  
  "phone": [], // 耳机音量增益取值范围  
}
```

请求方式

```
GET http://ip/usapi?method=set-volume&is-mic=1&mic-gain=0&is-spi=1&spi-gain=0&is-phone=1&phone-gain=0
```

参数	说明
method	方法名称: set-volume
is-mic	设置麦克风音量增益启用状态, 0: 静音 1: 启用
mic-gain	设置麦克风音量增益, 默认值: 0
is-spi	设置输入信号音量增益启用状态, 0: 静音 1: 启用
spi-gain	设置输入信号音量增益, 默认值: 0
is-phone	设置耳机音量增益启用状态, 0: 静音 1: 启用
phone-gain	设置耳机音量增益, 默认值: 0

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

set-video-color 接口

新增于: V1.3

设置图像的亮度、对比度、色调、饱和度。

通过 [get-info](#) 可以获取到录制相关参数的取值范围:

```
"color-range": {  
  "contrast": [],      // 图像对比度取值范围  
  "brightness": [],   // 图像亮度取值范围  
  "saturation": [],   // 图像饱和度取值范围  
  "hue": []           // 图像色调取值范围  
}
```

请求方式

```
GET http://ip/usapi?method=set-video-color&contrast=100&brightness=0&saturation=100&hue=0
```

参数	说明
method	方法名称: set-video-color
contrast	设置对比度, 默认值: 100
brightness	设置亮度, 默认值: 0
saturation	设置饱和度, 默认值: 100
hue	设置色调, 默认值: 0

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

set-low-latency-live 接口

新增于: V1.3

直播时是否开启低延时模式。

请求方式

```
GET http://ip/usapi?method=set-low-latency-live&is-low-latency=1
```

参数	说明
method	方法名称: set-low-latency-live
is-low-latency	是否开启 0: 不开启 1: 开启

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

set-auto-live 接口

新增于: V1.4

是否锁定直播, 当直播条件满足时自动开启直播, 此时无法手动关闭直播, 需先关闭锁定直播再关闭直播。

请求方式

```
GET http://ip/usapi?method=set-auto-live&is-auto-live=1
```

参数	说明
method	方法名称: set-auto-live
is-auto-live	是否锁定 0: 不锁定 1: 锁定

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

set-video-input-format 接口

新增于: V1.4

设置视频输入信号的色彩空间和量化范围。

通过 [get-info](#) 可以获取到录制相关参数的取值范围:

```
"video-format": {  
  "input-color-fmt": [],      // 色彩空间取值范围  
  "quant-range": [],        // 量化范围取值范围  
}
```

请求方式

```
GET http://ip/usapi?method=set-video-input-format&is-color-fmt=0&color-fmt=1&is-quant-range=0&quant-range=1
```

参数	说明
method	方法名称: set-video-input-format
is-color-fmt	是否自定义设置色彩空间, 默认值: 0
color-fmt	设置色彩空间
is-quant-range	是否自定义设置量化范围, 默认值: 0
quant-range	设置量化范围

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

set-video-output-format 接口

新增于: V1.4

设置视频输出的色彩空间和量化范围。

通过 [get-info](#) 可以获取到录制相关参数的取值范围:

```
"video-format": {  
  "output-color-fmt": [],      // 色彩空间取值范围  
  "quant-range": [],         // 量化范围取值范围  
  "sat-range": [],          // 饱和范围取值范围  
}
```

请求方式

```
GET http://ip/usapi?method=set-video-output-format&is-color-fmt=0&color-fmt=1&is-quant-range=0&quant-range=1&is-sat-range=0&sat-range=1
```

参数	说明
method	方法名称: set-video-output-format
is-color-fmt	是否自定义设置色彩空间, 默认值: 0
color-fmt	设置色彩空间, 默认值: 3
is-quant-range	是否自定义设置量化范围, 默认值: 0
quant-range	设置量化范围, 默认值: 2
is-sat-range	是否自定义设置饱和范围, 默认值: 0
sat-range	设置饱和范围, 默认值: 2

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

set-sync-offset 接口

新增于: V1.3

设置音频时延时间, 单位毫秒。音频滞后时, 建议设置为负值; 音频超前时, 建议设置为正值。

请求方式

```
GET http://ip/usapi?method=set-sync-offset&audio-sync-offset=100
```

参数	说明
method	方法名称: set-sync-offset
audio-sync-offset	设置音频时延时间, 取值范围为[-200,200]毫秒

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

add-nosignal-file 接口

新增于: V1.5

添加无输入信号时显示的图片, 仅支持分辨率小于 1920x1080, 文件大小小于 1MB 的 JPEG 图片, 最多支持添加 2 张。

请求方式

```
POST http://ip/usapi?method=add-nosignal-file
```

参数	说明
method	方法名称: add-nosignal-file

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

del-nosignal-file 接口

新增于: V1.5

删除用户添加的无信号图片，默认图片不可删除。

请求方式

```
GET http://ip/usapi?method=del-nosignal-file&id=0
```

参数	说明
method	方法名称: del-nosignal-file
id	图片 ID

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

set-nosignal-file 接口

新增于: V1.5

当启用无信号图片且设备无输入信号时, 显示选中的图片。

请求方式

```
GET http://ip/usapi?method=set-nosignal-file&id=0
```

参数	说明
method	方法名称: set-nosignal-file
id	图片 ID

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

use-nosignal-file 接口

新增于: V1.5

是否启用无信号图片。当设备无输入信号时，显示用户设备的图片。

请求方式

```
GET http://ip/usapi?method=use-nosignal-file&use-nosignal-file=0
```

参数	说明
method	方法名称: use-nosignal-file
use-nosignal-file	是否启用无信号图片 0: 不启用 1: 启用

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

start-format-usb 接口

对 U 盘进行格式化，操作后数据无法恢复，并且格式化过程不可取消。

进行格式化时，必须满足以下条件：

- 未进行固件升级
- 未进行重置和重启
- 未启用录制功能
- 未对 U 盘进行性能测试
- U 盘有效

通过 `get-status` 接口可以获取到以下信息：

```
"cur-status": 65552           // 设备运行状态掩码
"usb": {
  "disk-status": 1,          // U 盘状态掩码
  "total-size": 61638148096,
  "used-size": 97058816,
  ...
}
```

`cur-status` 和 [设备状态掩码](#) 进行与运算，可以得到以下状态表：

状态	条件
未进行固件升级	<code>cur-status & statusUpgrade != statusUpgrade</code>
未进行重置	<code>cur-status & statusReset != statusReset</code>
未进行重启	<code>cur-status & statusReboot != statusReboot</code>
未启用录制功能	<code>cur-status & statusRecord != statusRecord</code>
未对 U 盘进行性能测试	<code>cur-status & statusDiskTest != statusDiskTest</code>

`disk-status` 和 [U 盘状态码](#) 进行与运算，可以得到以下状态表：

状态	条件
U 盘有效	<code>disk-status & DiskValid = DiskValid</code>

请求方式

```
GET http://ip/usapi?method=start-format-usb
```

参数	说明
method	方法名称：start-format-usb

返回数据

```
{
  "result": 0
}
```

属性	说明
result	返回状态。 0: 开始格式化 1: 操作重复 -5: 磁盘错误 -9: 系统忙 返回其它值请参考 API 状态码

格式化过程状态

通过 `get-status` 接口可以获取到 `format-status` 信息，如下：

```
"format-status": {  
  "result": 27,  
  "percent": 20,  
  "client-id": ""  
}
```

属性	说明
result	操作状态。 27: 初始状态 2: 格式化中状态 0: 操作成功 返回其它值请参考 API 状态码
percent	格式化进度: 20 表示 20%
client-id	保留属性

U 盘格式化后, 需要调用 [clear-format-usb](#) 接口将格式化状态重置为初始状态 (27)。

start-format-sd 接口

新增于: V1.3

对 SD 卡进行格式化, 操作后数据无法恢复, 并且格式化过程不可取消。

进行格式化时, 必须满足以下条件:

- 未进行固件升级
- 未进行重置和重启
- 未启用录制功能
- SD 卡有效

通过 `get-status` 接口可以获取到以下信息:

```
"cur-status": 65552           // 设备运行状态掩码
"sd": {
  "disk-status": 1,           // SD 卡状态掩码
  "total-size": 61638148096,
  "used-size": 97058816,
  ...
}
```

`cur-status` 和 [设备状态掩码](#) 进行与运算, 可以得到以下状态表:

状态	条件
未进行固件升级	<code>cur-status & statusUpgrade != statusUpgrade</code>
未进行重置	<code>cur-status & statusReset != statusReset</code>
未进行重启	<code>cur-status & statusReboot != statusReboot</code>
未启用录制功能	<code>cur-status & statusRecord != statusRecord</code>

`disk-status` 和 [存储设备状态码](#) 进行与运算, 可以得到以下状态表:

状态	条件
SD 卡有效	<code>disk-status & DiskValid = DiskValid</code>

请求方式

```
GET http://ip/usapi?method=start-format-sd
```

参数	说明
method	方法名称: start-format-sd

返回数据

```
{
  "result": 0
}
```

属性	说明
result	返回状态。 0: 开始格式化 1: 操作重复 -5: 磁盘错误 -9: 系统忙 返回其它值请参考 API 状态码

格式化过程状态

通过 `get-status` 接口可以获取到 `format-status` 信息, 如下:

```
"format-status": {  
  "result": 27,  
  "percent": 20,  
  "client-id": ""  
}
```

属性	说明
result	操作状态。 27: 初始状态 2: 格式化中状态 0: 操作成功 返回其它值请参考 API 状态码
percent	格式化进度: 20 表示 20%
client-id	保留属性

SD 卡格式化后, 需要调用 [clear-format-sd](#) 接口将格式化状态重置为初始状态 (27)。

clear-format-usb 接口

U 盘进行格式化后，会处于格式化成功或失败的状态，调用该接口可以重置为初始化状态（27）。

请求方式

```
GET http://ip/usapi?method=clear-format-usb
```

参数	说明
method	方法名称：clear-format-usb

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0：操作成功，返回其它值请参考 API 状态码

clear-format-sd 接口

新增于: V1.3

SD 卡进行格式化后，会处于格式化成功或失败的状态，调用该接口可以重置为初始化状态 (27)。

请求方式

```
GET http://ip/usapi?method=clear-format-sd
```

参数	说明
method	方法名称: clear-format-sd

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 操作成功，返回其它值请参考 API 状态码

start-test-usb 接口

对 U 盘的读写性能进行测试，判断是否满足当前录制码率的写入速度要求。测试过程中可以调用 `stop-test-usb` 接口进行取消。

进行性能测时，必须满足以下条件：

- 未进行固件升级
- 未进行重置和重启
- 未启用录制功能
- 未对 U 盘进行格式化
- 未对 U 盘进行性能测试
- U 盘有效
- U 盘空间大于 200 M
- U 盘可写入

通过 `get-status` 接口可以获取到以下信息：

```
"cur-status": 65552           // 设备运行状态掩码
"usb": {
  "disk-status": 1,           // U 盘状态掩码
  "total-size": 61638148096,
  "used-size": 97058816,
  ...
}
```

`cur-status` 和 [设备状态掩码](#) 进行与运算，可以得到以下状态表：

状态	条件
未进行固件升级	<code>cur-status & statusUpgrade != statusUpgrade</code>
未进行重置	<code>cur-status & statusReset != statusReset</code>
未进行重启	<code>cur-status & statusReboot != statusReboot</code>
未启用录制功能	<code>cur-status & statusRecord != statusRecord</code>
未对 U 盘进行格式化	<code>cur-status & statusFormatDisk != statusFormatDisk</code>
未对 U 盘进行性能测试	<code>cur-status & statusDiskTest != statusDiskTest</code>

`disk-status` 和 [U 盘状态码](#) 进行与运算，可以得到以下状态表：

状态	条件
U 盘有效	<code>disk-status & DiskValid = DiskValid</code>
U 盘空间大于 200 M	<code>disk-status & LowSpace != LowSpace</code>
U 盘可写入	<code>disk-status & NotWrite != NotWrite</code>

请求方式

```
GET http://ip/usapi?method=start-test-usb
```

参数	说明
method	方法名称：start-test-usb

返回数据

```
{
  "result": 0
}
```

属性	说明
	返回状态。 0: 开始格式化 1: 操作重复

result	-5: 磁盘错误 -9: 系统忙 -16: 磁盘空间不足 返回其它值请参考 API 状态码
--------	--

计算最低写入速度 requirSpeed

通过 [get-settings](#) 接口可以获取到录制配置信息，如下：

```

"rec-stream": 0,
"main-stream": {
  "kbps": 4096,
  ...
},
"sub-stream": {
  "kbps": 1024,
  ...
}
"audio": {
  "kbps": 128
  ...
}

```

rec-stream 表示录制码流类型：0 为 **main-stream**，1 为 **sub-stream**

通过以上信息可计算出当前录制时，**requirSpeed** 为：

```

requirSpeed = main-stream.kbps + audio.kbps
             = (4096 Kbps + 128 Kbps)
             = 4224 Kbps / 1024 * 8
             ≈ 0.52 MB/S

```

性能测试过程状态

通过 [get-status](#) 接口可以获取到 **disk-test** 信息，如下：

```

"disk-test": {
  "read-bps": 0,
  "write-bps": 14833071, // 14833071 bps / 1024 * 1024 ≈ 14.15 MB/S
  "percent": 3,
  "result": 2,
  "client-id": "web-session-32435088"
}

```

属性	说明
result	操作状态。 27: 初始状态 5: 操作取消 2: 测试中状态 0: 操作成功 返回其它值请参考 API 状态码
percent	格式化进度：20 表示 20%
read-bps	当前读取速度
write-bps	当前写入速度，如果大于 requirSpeed 则说明 U 盘性能是达标的
client-id	保留属性

进行性能测试后，需要调用 [clear-test-usb](#) 接口将测试状态重置为初始状态（27）。

stop-test-usb 接口

调用 `start-test-usb` 接口后，U 盘会进入性能测试过程，可以调用该接口取消操作。

请求方式

```
GET http://ip/usapi?method=stop-test-usb
```

参数	说明
method	方法名称：stop-test-usb

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0：操作成功，返回其它值请参考 API 状态码

clear-test-usb 接口

U 盘通过 [start-test-usb](#) 进行性能测试，最后会处于测试成功或失败的状态，调用该接口可以重置为初始化状态（27）。

请求方式

```
GET http://ip/usapi?method=clear-test-usb
```

参数	说明
method	方法名称：clear-test-usb

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0：操作成功，返回其它值请参考 API 状态码

get-media-files 接口

新增于: V1.3

获取 U 盘 或 SD 卡中录制的文件。

获取到文件后, 可以调用 [del-media-files](#) 接口进行删除。

请求方式

```
GET http://ip/usapi?method=get-media-files&disk-type=1&start=0&count=3
```

参数	说明
method	方法名称: get-media-files
disk-type	存储设备类型, 0: U 盘, 1: SD 卡
start	起始索引, 从 0 开始
count	获取记录数量

返回数据

```
{
  "result": 0,
  "path": "/media/disk1/REC_Folder",
  "thumbnail": "thumbnail",
  "media-files": [
    {
      "name": "VID_9.mp4", // 视频文件名称
      "thumbnail-name": "VID_9.mp4_1569393321", // 缩略图名称, 默认后缀为 .jpg
      "status": 1, // 视频状态, 0 (正在录制), 1 (正常), 2 (错误), 3 (丢失)
      "create-time": "2019-09-25 06:35:21", // 创建时间
      "size-bytes": 4025670, // 视频大小, 单位: 字节
      "duration": 14480, // 视频时长, 单位: 秒
      "width": 1280, // 视频宽, 单位: 像素
      "height": 720, // 视频高, 单位: 像素
      "interval": 200000, // 帧率 (FPS) = 1000000 / interval
      "codec": 0 // 编码类型, 0: H264 1: HEVC
    },
    ...
  ]
}
```

属性	说明
result	返回状态。0: 操作成功, 返回其它值请参考 API 状态码
path	录制视频存储目录
thumbnail	视频缩率图存储子目录, 绝对路径为: path + thumbnail
media-files	文件列表

常用路径

名称	格式	示例
视频路径	path/name	/media/disk1/REC_Folder/VID_9.mp4
下载路径	http://ip:8080/download/path/name	http://ip:8080/download/media/disk1/REC_Folder/VID_9.mp4
图片路径	path/thumbnail/thumbnail-name	/media/disk1/REC_Folder/thumbnail/thumbnail-name.jpg

del-media-files 接口

新增于: V1.3

批量删除 U 盘 或 SD 卡中录制的文件。

请求方式

```
POST http://ip/usapi?method=del-media-files
```

参数	说明
method	方法名称: del-media-files

请求头

```
Content-Type: application/json;charset=UTF-8
```

Request Payload

```
{
  disk-type: 1,           // 存储设备类型, 0: U 盘, 1: SD 卡
  media-files: ["VID_6_14.mp4"] // 要删除的文件名称数组
}
```

返回数据

```
{
  "result": 0
}
```

属性	说明
result	返回状态。0: 操作成功, 返回其它值请参考 API 状态码

send-file-add-server 接口

新增文件上传服务器，最多可添加 8 条上传配置。

请求方式

```
GET http://ip/usapi?method=send-file-add-server&type=xxx&net-mode=1&...
```

参数	说明
method	方法名称: send-file-add-server
type	服务器类型 0: FTP/SFTP 1: Google Drive 2: Dropbox 3: YouTube
net-mode	网络优先级 0: 移动宽带优先 1: 有线网络优先 2: 无线网络优先
FTP/SFTP 上传参数	
proto	协议类型 0: FTP 1: SFTP
url	服务器地址
port	服务器端口
dir-name	目录名称
encryption	FTP 加密类型 0: 只使用明文 FTP 1: 要求显式的 FTP over TLS 2: 要求隐式的 FTP over TLS
login-type	登录类型 0: 匿名 1: 正常
user-name	用户名
passwd	密码
mode	传输模式 0: 主动 1: 被动
Google Drive 上传参数	
account-id	用户id
photo-path	用户头像地址
user-name	用户名
access-token	授权令牌
refresh-token	更新令牌
dir-name	目录名称
parents	父目录id
Dropbox 上传参数	
account-id	用户id
photo-path	用户头像地址
user-name	用户名
access-token	授权令牌
refresh-token	更新令牌
dir-name	目录名称
YouTube 上传参数	
account-id	用户id

photo-path	用户头像地址
user-name	用户名
access-token	授权令牌
refresh-token	更新令牌
privacy	权限

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 新增成功, 返回其它值请参考 API 状态码

send-file-del-server 接口

删除文件上传服务器。

请求方式

```
GET http://ip/usapi?method=send-file-del-server&id=0
```

参数	说明
method	方法名称: send-file-del-server
id	服务器 ID

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 删除成功, 返回其它值请参考 API 状态码

send-file-enable-server 接口

首个新增服务器默认是启用状态，同一时间只支持向一个服务器上传文件。

请求方式

```
GET http://ip/usapi?method=send-file-enable-server&id=1&is-use=1
```

参数	说明
method	方法名称: send-file-enable-server
id	服务器 ID
is-use	启用状态 0: 禁用 1: 启用

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 启用成功, 返回其它值请参考 API 状态码

send-file-set-server 接口

修改文件上传服务器。

请求方式

```
GET http://ip/usapi?method=send-file-set-server&id=xxx&type=xxx&net-mode=1&...
```

参数	说明
method	方法名称: send-file-set-server
id	服务器 ID
type	服务器类型 0: FTP/SFTP 1: Google Drive 2: Dropbox 3: YouTube
net-mode	网络优先级 0: 移动宽带优先 1: 有线网络优先 2: 无线网络优先
FTP/SFTP 上传参数	
proto	协议类型 0: FTP 1: SFTP
url	服务器地址
port	服务器端口
dir-name	目录名称
encryption	FTP 加密类型 0: 只使用明文 FTP 1: 要求显式的 FTP over TLS 2: 要求隐式的 FTP over TLS
login-type	登录类型 0: 匿名 1: 正常
user-name	用户名
passwd	密码
mode	传输模式 0: 主动 1: 被动
Google Drive 上传参数	
account-id	用户 ID
photo-path	用户头像地址
user-name	用户名
access-token	授权令牌
refresh-token	更新令牌
dir-name	目录名称
parents	父目录 ID
Dropbox 上传参数	
account-id	用户 ID
photo-path	用户头像地址
user-name	用户名
access-token	授权令牌
refresh-token	更新令牌
dir-name	目录名称
YouTube 上传参数	

account-id	用户 ID
photo-path	用户头像地址
user-name	用户名
access-token	授权令牌
refresh-token	更新令牌
privacy	权限

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 修改成功, 返回其它值请参考 API 状态码

send-file-start-test-server 接口

对服务器进行上传测试，测试过程中可以调用 [send-file-stop-test-server](#) 接口进行取消，目前仅支持 FTP/SFTP 服务器上传测试。

请求方式

```
GET http://ip/usapi?method=send-file-start-test-server&type=xxx&net-mode=1&...
```

参数	说明
method	方法名称: send-file-start-test-server
type	服务器类型 0: FTP/SFTP 1: Google Drive 2: Dropbox 3: YouTube
net-mode	网络优先级 0: 移动宽带优先 1: 有线网络优先 2: 无线网络优先
FTP/SFTP 上传参数	
proto	协议类型 0: FTP 1: SFTP
url	服务器地址
port	服务器端口
dir-name	目录名称
encryption	FTP 加密类型 0: 只使用明文 FTP 1: 要求显式的 FTP over TLS 2: 要求隐式的 FTP over TLS
login-type	登录类型 0: 匿名 1: 正常
user-name	用户名
passwd	密码
mode	传输模式 0: 主动 1: 被动

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 新增成功, 返回其它值请参考 API 状态码

send-file-stop-test-server 接口

调用 [send-file-start-test-server](#) 接口后，服务器会进入上传测试过程，可以调用该接口取消操作。

请求方式

```
GET http://ip/usapi?method=send-file-stop-test-server
```

参数	说明
method	方法名称: send-file-stop-test-server

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 新增成功, 返回其它值请参考 API 状态码

clear-test-send-file 接口

通过 [send-file-start-test-server](#) 进行上传测试，最后会处于测试成功或失败的状态，调用该接口可以重置为初始化状态 (27)。

请求方式

```
GET http://ip/usapi?method=clear-test-send-file
```

参数	说明
method	方法名称: clear-test-send-file

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

send-file-add 接口

添加文件至文件上传列表，目前只支持上传录制在 SD 卡的文件。

请求方式

```
POST http://ip/usapi?method=send-file-add
```

参数	说明
method	方法名称: send-file-add

请求头

```
Content-Type: application/json;charset=UTF-8
```

Request Payload

```
{
  disk-type: 1,           // 存储设备类型, 0: U 盘, 1: SD 卡
  media-files: ["VID_6_14.mp4"] // 要删除的文件名称数组
}
```

返回数据

```
{
  "result": 0
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

send-file-del 接口

从文件上传列表删除文件。

请求方式

```
POST http://ip/usapi?method=send-file-del
```

参数	说明
method	方法名称: send-file-del

请求头

```
Content-Type: application/json;charset=UTF-8
```

Request Payload

```
{
  disk-type: 1,           // 存储设备类型, 0: U 盘, 1: SD 卡
  media-files: ["VID_6_14.mp4"] // 要删除的文件名称数组
}
```

返回数据

```
{
  "result": 0
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

send-file-clear 接口

清除文件列表。

请求方式

```
GET http://ip/usapi?method=send-file-clear
```

参数	说明
method	方法名称: send-file-clear

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

send-file-get-status 接口

获取文件上传列表，文件上传进度等。

请求方式

```
GET http://ip/usapi?method=send-file-get-status&disk-type=1&start=0&count=0
```

参数	说明
method	方法名称: send-file-get-status
disk-type	存储设备类型, 0: U 盘, 1: SD 卡
start	起始索引, 从 0 开始
count	获取记录数量

返回数据

JSON 结构如下:

```
{
  "result": 0,                // 返回状态
  "total-count-ongoing": 1,   // 待上传文件数目
  "total-count-done": 2,     // 已上传文件数目
  "start": 0,
  "count": 10,
  "status": {},              // 当前文件上传状态
  "media-files": {},        // 文件上传列表
}
```

属性	说明
result	返回状态, 0: 获取状态成功, 返回其它值请参考 API 状态码

send-file-set-is-auto 接口

是否自动将录制在 SD 卡的文件添加至文件上传列表，默认为手动添加。

请求方式

```
GET http://ip/usapi?method=send-file-set-is-auto&is-auto=0
```

参数	说明
method	方法名称：send-file-set-is-auto
is-auto	是否自动 0: 手动 1: 自动

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 操作成功，返回其它值请参考 API 状态码

send-file-start 接口

启动文件上传，默认为启动状态。

请求方式

```
GET http://ip/usapi?method=send-file-start
```

参数	说明
method	方法名称：send-file-start

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0：操作成功，返回其它值请参考 API 状态码

send-file-stop 接口

停止文件上传。

请求方式

```
GET http://ip/usapi?method=send-file-stop
```

参数	说明
method	方法名称: send-file-stop

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

add-server 接口

新增直播服务器，最多可同时录入16条直播配置。

配置相同的直播仅可录入一个：

- 自定义类型 RTMP 直播，URL 和 StreamKey 都相同时为同一配置
- 其它类型 RTMP 直播，StreamKey 相同时为同一配置
- RTSP 直播，端口号相同时为同一配置

请求方式

```
GET http://ip/usapi?method=add-server&type=xxx&url=xxx&key=xxx&...
```

参数	说明
method	方法名称：add-server
type	服务器类型 0: RTMP 1: Twitch 2: YouTube 3: Facebook 100: RTSP
name	服务器名称，字符要求： 1、1 - 32 个字符 2、由 A-Z, a-z, 0-9, 空格 . _ - + [] () 组成，并且不能以空格开头或结束
RTMP直播参数	
url	服务器地址
key	流密钥
is-auth	认证状态 0: 无需认证 1: 需要认证，用户名和密码必填
user	用户名
passwd	密码
token	Token
event-data	直播事件，仅用于 YouTube 和 Facebook 直播
net-mode	网络优先 0: 移动宽带优先 1: 有线网络优先 2: 无线网络优先
RTSP直播参数	
port	端口号，默认为554
is-main	是否启用主码流 0: 不启用 1: 启用
main-stream-name	主码流名称，字符要求： 1、1 - 32 个字符 2、由 A-Z, a-z, 0-9, 空格 . _ - + [] () 组成，并且不能以空格开头或结束
is-sub	是否启用次码流 0: 不启用 1: 启用
sub-stream-name	次码流名称，字符要求： 1、1 - 32 个字符 2、由 A-Z, a-z, 0-9, 空格 . _ - + [] () 组成，并且不能以空格开头或结束
is-audio	是否启用音频 0: 不启用 1: 启用

返回数据

```
{
```

```
"result": 0  
}
```

属性	说明
result	返回状态, 0: 新增成功, 返回其它值请参考 API 状态码

enable-server 接口

新增服务器默认是禁用状态，只有通过该接口启用之后，才可以进行直播推流。

默认情况下最大支持启用两路直播，但有以下情况时，只允许启用一路：

- 当直播使用的编码器码率大于 4 Mbps 时
- 当直播是 RTSP 直播时

请求方式

```
GET http://ip/usapi?method=enable-server&id=1&is-use=1
```

参数	说明
method	方法名称：enable-server
id	服务器 ID
is-use	启用状态 0: 禁用 1: 启用

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态，0：新增成功，返回其它值请参考 API 状态码

set-server 接口

修改直播服务器。

请求方式

```
GET http://ip/usapi?method=set-server&id=xxx&type=xxx&url=xxx&key=xxx&...
```

参数	说明
method	方法名称: set-server
id	服务器 ID
type	服务器类型 0: RTMP 1: Twitch 2: YouTube 3: Facebook 100: RTSP
name	服务器名称, 字符要求: 1、1 - 32 个字符 2、由 A-Z, a-z, 0-9, 空格 ._-+'[]() 组成, 并且不能以空格开头或结束
RTMP直播参数	
url	服务器地址
key	流密钥
is-auth	认证状态 0: 无需认证 1: 需要认证, 用户名和密码必填
user	用户名
passwd	密码
token	Token
event-data	直播事件, 仅用于 YouTube 和 Facebook 直播
net-mode	网络优先 0: 移动宽带优先 1: 有线网络优先 2: 无线网络优先
RTSP直播参数	
port	端口号, 默认为554
is-main	是否启用主码流 0: 不启用 1: 启用
main-stream-name	主码流名称, 字符要求: 1、1 - 32 个字符 2、由 A-Z, a-z, 0-9, 空格 ._-+'[]() 组成, 并且不能以空格开头或结束
is-sub	是否启用次码流 0: 不启用 1: 启用
sub-stream-name	次码流名称, 字符要求: 1、1 - 32 个字符 2、由 A-Z, a-z, 0-9, 空格 ._-+'[]() 组成, 并且不能以空格开头或结束
is-audio	是否启用音频 0: 不启用 1: 启用

返回数据

```
{  
  "result": 0  
}
```

属性	说明
----	----

result	返回状态, 0: 新增成功, 返回其它值请参考 API 状态码
--------	---

del-server 接口

删除直播服务器。

请求方式

```
GET http://ip/usapi?method=del-server&id=1
```

参数	说明
method	方法名称: del-server
id	服务器 ID

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 新增成功, 返回其它值请参考 API 状态码

start-test-server 接口

对服务器进行推流测试，同时间仅可对一路直播进行测试，测试过程中可以调用 [stop-test-server](#) 接口进行取消；当测试时间达到 20 秒时，自动停止直播测试。

请求方式

```
GET http://ip/usapi?method=start-test-server&type=xxx&url=xxx&key=xxx&...
```

参数	说明
method	方法名称: start-test-server
type	服务器类型 0: RTMP 1: Twitch 2: YouTube 3: Facebook
url	服务器地址
key	流密钥
is-auth	认证状态 0: 无需认证 1: 需要认证, 用户名和密码必填
user	用户名
passwd	密码
token	Token
net-mode	网络优先 0: 移动宽带优先 1: 有线网络优先 2: 无线网络优先

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 开始测试 1: 操作重复 -9: 系统忙 返回其它值请参考 API 状态码

测试过程状态

通过 [get-status](#) 接口可以获取到 **living-test** 信息，如下：

```
"living-test": {  
  "result": 27,  
  "upload-bps": 0,  
  "percent": 0,  
  "net": 0,  
  "client-id": ""  
}
```

属性	说明
result	操作状态。 27: 初始状态 5: 操作取消 2: 测试中状态 0: 操作成功 返回其它值请参考 API 状态码

percent	测试进度：20 表示 20%
upload-bps	上传速度
net	网络类型
client-id	保留属性

进行测试后，需要调用 [clear-test-server](#) 接口将测试状态重置为初始状态（27）。

stop-test-server 接口

调用 [start-test-server](#) 接口后，服务器会进入推流测试过程，可以调用该接口取消操作。

请求方式

```
GET http://ip/usapi?method=stop-test-server
```

参数	说明
method	方法名称: stop-test-server

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 操作成功，返回其它值请参考 API 状态码

clear-test-server 接口

通过 [start-test-server](#) 进行推流测试，最后会处于测试成功或失败的状态，调用该接口可以重置为初始化状态 (27)。

请求方式

```
GET http://ip/usapi?method=clear-test-server
```

参数	说明
method	方法名称: clear-test-server

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

set-stream 接口

选择任务使用的编码流。通常主编码流的性能优于次编码流。

请求方式

```
GET http://ip/usapi?method=set-stream&rec=0&live=1&mws=1
```

参数	说明
method	方法名称: set-stream
rec	录制编码流 0: 主编码流 1: 次编码流
live	直播编码流 0: 主编码流 1: 次编码流
mws	App 客户端预览码流 0: 主编码流 1: 次编码流

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

set-video 接口

配置主编码流参数和次编码流参数，相关参数如下：

- 视频分辨率
- 帧间隔
- 视频编码类型：主编码流可以选择 H264 或 HEVC，次编码流默认必须选择 H264。
- 编码质量
- 视频码率
- 关键帧间隔
- 量化范围
- 位率统计周期

通过 `get-info` 可以获取到各个参数的取值范围：

```
{
  "codec-cap": {
    "resolutions": [], // 视频分辨率取值范围
    "durations": [], // 帧间隔取值范围
    "video-codec": [], // 视频编码类型取值范围
    "profile": [], // H264 编码质量取值范围
    "hevc-profile": [], // HEVC 编码质量取值范围
    "video-kbps": [], // 视频编码码率取值范围
    "gop-sec": [], // 关键帧间隔取值范围
    "video-range": [], // 量化范围取值范围
    "stat-sec": [] // 位率统计周期取值范围
  }
}
```

请求方式

```
GET http://ip/usapi?method=set-video&stream=0&is-auto=0&cx=1280&cy=720&duration=333333&kbps=1024&gop=1&fourcc=0&profile=2&cbrstat=60&fullrange=1&codec=0
```

参数	说明
method	方法名称：set-video
stream	编码流类型 0：主编码流 1：次编码流
is-auto	设置方式，配置次编码流时只能为0 0：手动 1：自动
cx	分辨率宽度像素值
cy	分辨率高度像素值
duration	帧间隔
kbps	视频编码码率
gop	关键帧间隔
fourcc	保留参数。默认值：0
codec	编码类型
profile	编码质量
cbrstat	位率统计周期
fullrange	量化范围

返回数据

```
{
  "result": 0
}
```

属性	说明
result	返回状态。0：操作成功，返回其它值请参考 API 状态码

set-audio 接口

配置音频编码参数。

通过 [get-info](#) 可以获取到音频编码码率取值范围：

```
{
  "codec-cap": {
    "audio-kbps": [],          // 音频编码码率取值范围
  }
}
```

请求方式

```
GET http://ip/usapi?method=set-audio&kbps=48
```

参数	说明
method	方法名称：set-audio
kbps	音频编码码率

返回数据

```
{
  "result": 0
}
```

属性	说明
result	返回状态，0：操作成功，返回其它值请参考 API 状态码

reboot 接口

重启设备，重启后需要重新登录。

重启过程大概需要几分钟时间，可以使用 [ping 接口](#) 判断设备是否已经重启。

请求方式

```
GET http://ip/usapi?method=reboot
```

参数	说明
method	方法名称: reboot

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0 操作成功, 设备进入重启状态。返回其它值请参考 API 状态码

reset-all-settings 接口

将设备全部参数恢复至默认值。

重置时重新初始化网络，整个过程大概需要几分钟时间。可以使用 [ping 接口](#) 判断设备是否已经重置完成。

请求方式

```
GET http://ip/usapi?method=reset-all-settings
```

参数	说明
method	方法名称: reset-all-settings

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

set-enable-key-reset 接口

新增于: V1.4

是否允许通过长按直播按键 5 秒将设备配置参数恢复至默认值。

请求方式

```
GET http://ip/usapi?method=set-enable-key-reset&enable-key-reset=0
```

参数	说明
method	方法名称: set-enable-key-reset
enable-key-reset	是否允许 0: 否 1: 是

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

set-enable-key-reboot 接口

新增于: V1.4

是否允许通过长按录制按键 5 秒重新启动设备。

请求方式

```
GET http://ip/usapi?method=set-enable-key-reboot&enable-key-reboot=0
```

参数	说明
method	方法名称: set-enable-key-reboot
enable-key-reboot	是否允许 0: 否 1: 是

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

login 接口

用户登录，登录成功后会在 Cookie 存放 Session ID (Cookie: sid=e0f6b33dd2b575eff40733b3778beaab)。

请求方式

```
GET http://ip/usapi?method=login&id=xxx&pass=xxx
```

参数	说明
method	方法名称: login
id	用户名
pass	密码, MD5 加密

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 登录成功, 36: 用户名或密码错误, 返回其它值请参考 API 状态码

logout 接口

退出登录, 返回到登录界面。

请求方式

```
GET http://ip/usapi?method=logout
```

参数	说明
method	方法名称: logout

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 退出登录成功, 返回其它值请参考 API 状态码

get-users 接口

获取系统用户列表信息，仅管理员有权限。

请求方式

```
GET http://ip/usapi?method=get-users
```

参数	说明
method	方法名称: get-users

返回数据

```
{
  "result": 0,
  "users": [
    {
      "id": "Admin",
      "type": 1
    },
    {
      "id": "Test",
      "type": 2
    }
  ]
}
```

属性	说明
result	返回状态, 0: 数据获取成功, 返回其它值请参考 API 状态码
users	用户列表数组, id: 用户名, type: 类型, 1 为管理员

add-user 接口

添加用户，仅管理员有权限。

请求方式

```
GET http://ip/usapi?method=add-user&id=xxx&pass=xxx
```

参数	说明
method	方法名称: add-user
id	用户名
pass	密码, MD5 加密

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 新增成功, 返回其它值请参考 API 状态码

del-user 接口

删除用户，仅管理员有权限。

请求方式

```
GET http://ip/usapi?method=del-user&id=xxx
```

参数	说明
method	方法名称: del-user
id	用户登录名

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 删除成功, 返回其它值请参考 API 状态码

ch-password 接口

用户修改自己的登录密码，修改时必须输入原密码。

请求方式

```
GET http://ip/usapi?method=ch-password&pass=xxx&new-pass=xxx
```

参数	说明
method	方法名称: ch-password
pass	原密码, MD5 加密
new-pass	新密码, MD5 加密

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 修改成功, 返回其它值请参考 API 状态码

set-password 接口

重置用户密码，无需输入原密码，仅管理员有权限。

请求方式

```
GET http://ip/usapi?method=set-password&id=xxx&pass=xxx
```

参数	说明
method	方法名称: set-password
id	用户登录名
pass	新密码, MD5 加密

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 重置成功, 返回其它值请参考 API 状态码

set-net 接口

设置以太网和 Wi-Fi 配置信息。

请求方式

```
GET http://ip/usapi?method=set-net&type=0&is-dhcp=1&ip=10.10.107.212&mask=255.255.0.0&router=10.10.0.1&dns=10.0.0.3
```

参数	说明
method	方法名称: set-net
type	网口类型 0: 无线网卡 1: 有线网卡
is-dhcp	是否动态获取 IP 地址 0: 手动设置 1: 动态获取
ip	IP 地址
mask	子网掩码
router	网关
dns	DNS 服务器

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

disconn-wifi 接口

断开无线网络连接。

请求方式

```
GET http://ip/usapi?method=disconn-wifi
```

参数	说明
method	方法名称: disconn-wifi

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

close-softap 接口

新增于: V1.3

开启 AP 模式。

请求方式

```
GET http://ip/usapi?method=open-softap
```

参数	说明
method	方法名称: open-softap

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

close-softap 接口

新增于: V1.3

关闭 AP 模式。

请求方式

```
GET http://ip/usapi?method=close-softap
```

参数	说明
method	方法名称: close-softap

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

update 接口

固件更新有两种方式：

- 手动更新：用户可以先通过 [upload-update-file](#) 接口上传指定版本的固件，然后再进行更新
- 在线升级：当有新版本时，用户可以在线升级到最新版本，[online-update-check](#) 接口检测线上是否有新版本固件。

进行固件更新时，必须满足以下条件：

- 未进行 U 盘格式化
- 未进行检测新版本固件

通过 [get-status](#) 接口可以获取到以下信息：

```
"cur-status": 65552 // 设备运行状态掩码
```

`cur-status` 和 [设备状态掩码](#) 进行与运算，可以得到以下状态表：

状态	条件
未对 U 盘进行格式化	<code>cur-status & statusFormatDisk != statusFormatDisk</code>
未进行检测新版本固件	<code>cur-status & statusCheckUpgrade != statusCheckUpgrade</code>

请求方式

```
GET http://ip/usapi?method=update&mode=xxx
```

参数	说明
method	方法名称：update
mode	更新方式 upload：手动更新 online：在线升级

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态，0：开始更新，返回其它值请参考 API 状态码

升级过程状态

通过 [get-status](#) 接口可以获取到 `upgrade-status` 信息，如下：

```
"upgrade-status": {  
  "result": 27,  
  "step": 0,  
  "percent": 0,  
  "mode": "none",  
  "client-id": ""  
}
```

属性	说明
result	操作状态 27：初始状态 2：更新中状态 0：操作成功 返回其它值请参考 API 状态码
step	更新步骤

percent	格式化进度: 20 表示 20%
mode	更新方式
client-id	保留属性

如果更新失败, 需要调用 [clear-upgrade](#) 接口将更新状态重置为初始状态 (27)。

upload-update-file 接口

上传固件，上传文件格式为.mwf。

请求方式

```
POST http://ip/usapi?method=upload-update-file
```

参数	说明
method	方法名称: upload-update-file

返回数据

```
{
  "status": 0,
  "up-to-date": true,
  "version": "1.1.72",
  "size": 11890776
}
```

属性	说明
status	返回状态, 0: 上传成功, 返回其它值请参考 API 状态码
up-to-date	是否为最新版本, true/false
version	上传固件版本号
size	上传固件大小 (B)

固件上传完成后, 可以调用 [update](#) 接口进行更新。

cancel-download 接口

固件通过 [update](#) 进行在线升级时，会先下载新固件，下载过程可以调用该接口进行取消。

请求方式

```
GET http://ip/usapi?method=cancel-download
```

参数	说明
method	方法名称：cancel-download

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0：操作成功，返回其它值请参考 API 状态码

online-update-check 接口

在线检测是否有新版本的固件。

进行检测时，必须满足以下条件：

- 未进行固件升级

通过 `get-status` 接口可以获取到以下信息：

```
"cur-status": 65552 // 设备运行状态掩码
```

`cur-status` 和 [设备状态掩码](#) 进行与运算，可以得到以下状态表：

状态	条件
未进行固件升级	<code>cur-status & statusUpgrade != statusUpgrade</code>

请求方式

```
GET http://ip/usapi?method=online-update-check
```

参数	说明
method	方法名称：online-update-check

返回数据

```
{
  "result": 0
}
```

属性	说明
result	返回状态，0：操作成功，返回其它值请参考 API 状态码

检测过程状态

通过 `get-status` 接口可以获取到 `check-upgrade` 和 `upgrade` 信息，如下：

```
"check-upgrade": {
  "result": 0,
  "client-id": ""
},
"upgrade": {
  "ver": "1.2.123",
  "date": "2012-1-1 00:00:00",
  "size-byte": 12004784,
  "info": [
    {
      "version": "1.2.123",
      "changelog": "## Develop version 1.2. ## Develop version2."
    }
  ]
}
```

属性	说明
result	操作状态。 27：初始状态 2：检测中状态 0：操作成功 返回其它值请参考 API 状态码
ver	新固件版本号
date	新固件发布时间

size-byte	新固件文件大小 (B)
info	更新日志
client-id	保留属性

检测操作失败后，需要调用 [clear-check-update](#) 接口将检测状态重置为初始状态 (27)。

clear-upgrade 接口

固件通过 [update](#) 进行手动更新或在线升级后，如果更新失败，调用该接口可以重置为初始化状态（27）。

请求方式

```
GET http://ip/usapi?method=clear-upgrade
```

参数	说明
method	方法名称：clear-upgrade

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0：操作成功，返回其它值请参考 API 状态码

clear-check-update 接口

通过 [online-update-check](#) 进行检测新版本固件，最后会处于检测成功或失败的状态，调用该接口可以重置为初始化状态（27）。

请求方式

```
GET http://ip/usapi?method=clear-check-update
```

参数	说明
method	方法名称: clear-check-update

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 操作成功，返回其它值请参考 API 状态码

set-enable-check-update 接口

新增于: V1.4

是否允许设备进行固件版本检测。

请求方式

```
GET http://ip/usapi?method=set-enable-check-update&is-check-update=1
```

参数	说明
method	方法名称: set-enable-check-update
is-check-update	是否允许 0: 不允许 1: 允许

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

import-edid 接口

为输入端口导入 EDID 文件。

请求方式

```
POST http://ip/usapi?method=import-edid
```

参数	说明
method	方法名称: import-edid

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 API 状态码

export-edid 接口

将输入端口 EDID 保存到本地。

请求方式

```
GET http://ip/usapi?method=export-edid&file-name=...
```

参数	说明
method	方法名称: export-edid
file-name	保存的文件路径

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

set-default-edid 接口

将输入端口 EDID 恢复到默认值。

请求方式

```
GET http://ip/usapi?method=set-default-edid
```

参数	说明
method	方法名称: set-default-edid

返回数据

```
{
  "result": 0
  "data": "AP////////wA09wEAAQAAAAEaAQOAAAB4Au6Vo1RMmSYPUFT//4AxQEVAYUBxQIGA0QDhwAEAC0gAMPJwWoCwWIoAUB10AAAEAjqAG
HE4LUBYLEUAUB10AAAEAAAA/QAP1g+HPAAAAAAAAAAAAAAAA/ABNQUdFV0VMTAogICAgAWYCA1HxV2EQHwQTBRQgISJdX19gZWZiY2QHFgMSMgl/B
xUHUD0GwFcGAF9/AWd/AINPAADiAA9uAwwAEAC4eCEQgAECaWrn2F3EAXiAA+MPAeABHYAYcRwWIFgsJQBAhGMAAJ5mIVaqUQAeMEaPMwBQHxQAA
B4AAAAAAAAAAAAzw=="
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

get-edid-config 接口

获取输入端口 EDID 配置信息。

请求方式

```
GET http://ip/usapi?method=get-edid-config
```

参数	说明
method	方法名称: get-edid-config

返回数据

```
{
  "result": 0
  "smart-edid": 1,
  "keep-last": 0,
  "add-audio": 1,
  "limit-pixel-clock": 1,
  "data": "AP////////wA09wEAAQAAAAEaAQOAAAB4Au6Vo1RMmSYPUFT//4AxQEVAyUBxQIGA0QDhwAEAC0gAMPJwWoCwWIoAUB10AAAEAjqAG
HE4LUBYLEUAUB10AAAEAAAA/QAPlg+HPAAAAAAAAAAAAAAAA/ABNQUdFV0VMTAogICAgAWYCA1HxV2EQHwQTBRQgISJdX19gZWZiY2QHFgMSMgl/B
xUHUD0GwFcGAF9/Awd/AINPAADiAA9uAwwAEAC4eCEQgAECaWrn2F3EAXiAA+MPAeABHYAYcRwWIFgsJQBAhGMAAJ5mIVaqUQAeMEaPMwBQHxQAA
B4AAAAAAAAAAAAAAzw=="
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

set-edid-config 接口

修改输入端口 EDID 配置信息。

请求方式

```
GET http://ip/usapi?method=set-edid-config&...
```

参数	说明
method	方法名称: set-edid-config
smart-edid	是否启用 SmartEDID 0: 否 1: 是
keep-last	是否保持最后一次的环出 EDID 0: 否 1: 是
add-audio	是否加入音频 0: 否 1: 是
limit-pixel-clock	是否使用限制像素时钟 0: 否 1: 是

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

get-loop-through-edid 接口

获取环出端口 EDID 信息。

请求方式

```
GET http://ip/usapi?method=get-loop-through-edid
```

参数	说明
method	方法名称: get-loop-through-edid

返回数据

```
{
  "result": 0
  "data": "AP////////wA09wEAAQAAAAEaAQOAAAB4Au6Vo1RMmSYPUFT//4AxQEVAYUBxQIGA0QDhwAEAC0gAMPJwWoCwWIoAUB10AAAEAjqAG
HE4LUBYLEUAUB10AAAEAAAA/QAP1g+HPAAAAAAAAAAAAAAAA/ABNQUdFV0VMTAogICAgAWYCA1HxV2EQHwQTBRQgISJdX19gZWZiY2QHFgMSMgl/B
xUHUD0GwFcGAF9/AWd/AINPAADiAA9uAwwAEAC4eCEQgAECARn2F3EAXiAA+MPAeABHYAYcRwWIFgsJQBAhGMAAJ5mIVaqUQAeMEaPMwBQHxQAA
B4AAAAAAAAAAAAzw=="
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

export-loop-through-edid 接口

将环出端口 EDID 保存到本地。

请求方式

```
GET http://ip/usapi?method=export-loop-through-edid&file-name=...
```

参数	说明
method	方法名称: export-loop-through-edid
file-name	保存的文件路径

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态。 0: 操作成功, 返回其它值请参考 API 状态码

cloud-reg-ex 接口

新增于: V1.5

注册到云平台, 默认可以注册到 2 个云平台服务器。

请求方式

```
GET http://ip:8070/cloud-api?method=cloud-reg-ex&id=1&cloud-enable-https=0&...
```

参数	说明
method	方法名称: cloud-reg-ex
id	云平台序号, 有效值: 0 或 1
cloud-code	邀请码, 4 位数字组成的字符串
cloud-ip-addr	服务器地址
cloud-http-port	http 服务器端口
cloud-enable-https	是否启用 https 服务 0: 不启用 1: 启用
cloud-https-port	https 服务器端口

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 云平台 API 状态码

cloud-unreg-ex 接口

新增于: V1.5

取消注册到云平台。

请求方式

```
GET http://ip:8070/cloud-api?method=cloud-unreg-ex&id=1
```

参数	说明
method	方法名称: cloud-reg-ex
id	云平台序号, 有效值: 0 或 1

返回数据

```
{  
  "result": 0  
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 云平台 API 状态码

cloud-status 接口

新增于: V1.5

获取云平台服务器状态。

请求方式

GET http://ip:8070/cloud-api?method=cloud-status&version=1

参数	说明
method	方法名称: cloud-status
version	云平台版本, 有效值: 1

返回数据

```
{
  "device_id": "A305200908004", // 设备序列号
  "number": 2, // 支持云平台数量
  "version": 1,
  "result": 0,
  "status": [
    {
      "cloud-code": "",
      "cloud-date": 0,
      "cloud-enable-https": 0,
      "cloud-http-port": 80,
      "cloud-https-port": 443,
      "cloud-ip-addr": "10.0.1.32",
      "cloud-reg-status": 101,
      "cloud-status": 35,
      "id": 0,
      "is-cloud-set": 1
    },
    {
      "cloud-code": "",
      "cloud-date": 0,
      "cloud-enable-https": 0,
      "cloud-http-port": 80,
      "cloud-https-port": 443,
      "cloud-ip-addr": "10.10.8.233",
      "cloud-reg-status": 103,
      "cloud-status": 35,
      "id": 1,
      "is-cloud-set": 1
    }
  ]
}
```

属性	说明
result	返回状态, 0: 操作成功, 返回其它值请参考 云平台 API 状态码