

E3 WEB API 说明书 (Version 1.21.0008)

调用示例:

* GET

```
curl -X GET "http://127.0.0.1/api/systemctrl/users/auth_info" \  
-H "authorization: eyJ0eXAiOi..." # 替换为有效 Token
```

* POST

```
curl -X POST "http://127.0.0.1/api/systemctrl/users/auth_info" \  
-H "Content-Type: application/json" \  
-H "authorization: eyJ0eXAiOi..." \  
-d '{"enable": false}'
```

1 系统服务/Service of systemctrl

1.1 用户管理/Users

1.1.1 用户登录

* 接口描述

单独请求该接口并不能直接登录，需要拿着接口下返回的 token 和 alias 值，结合 username 构造 cookie，并将其加入到请求头中请求 1.1.2 的校验接口，校验通过后才能登录。

* 请求地址

/api/systemctrl/users/login

* 请求方法

POST

* 请求参数

```
{  
  "username": "admin",  
  "password": "Admin123"  
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
App	{"language": "en"}	Y	{"language": "en"}	请求头参数，目前版本只支持中英文语言，分别为"en"/"zh"，用于弹窗提示的多语言输出。 **注意: **当前版本中，如果单独

				请求此接口时想要返回正确的错误提示，请务必在请求头中加入此参数。否则返回的提示中 msg 字段将为 null
username	String	Y	admin	用户名
password	String	Y	Admin123	密码

* 响应参数

```
{
  "result": "ok",
  "data": {
    "token": "0621da14f9d6546688cf87c5eab9c2f9",
    "alias": "Admin",
    "changed": true
  },
  "msg": null
}
```

* 响应参数说明

字段名	字段类型	字段描述
token	String	登录签发的 token，用于登录保持
alias	String	别名
changed	Boolean	用户是否已修改初始密码。未修改初始密码，前端跳转修改初始密码页

1.1.2 登录校验

* 接口描述

用于正常登录请求，和 kilolink 反向代理过来请求两方面的 session 校验和验证放行。正常登录请求需要在在请求头中添加 cookie 并组装字段值，参考样例；kilolink 反向代理的则需要在请求中添加 platform 字段，并指定相应的字段标识符，参考样例。

注意：1.21 及以后的版本增加了 API 鉴权开关，若开启该开关，则每个接口都会额外校验 Authorization 字段，需要在请求头参数加入该字段，否则接口会返回 Unauthorized。

* 请求地址

/api/systemctrl/users/session

* 请求方法

GET

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
platform	String	Y	"this-is-made-by-kiloview-for-platform-login"	请求参数，如果想单独使用此接口，需要用户将该值单独加入到 kilolink server 端请求头中。
Cookie	String	Y	"language=zh; Lang=zh-CN; username=admin; user=admin; alias=Admin; token=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ1c2VybmFtZSI6ImFkbWludiwibWQ1X3N0cil6ImVhMWI3MWI5YTljMWFkNjAwNW11"	请求头参数，如果想单独使用此接口，需要用户将该值单独加入到请求头中。
Authorization	String	Y	eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ1c2VybmFtZSI6ImFkbWludiwibWQ1X3N0cil6ImVhMWI3MWI5YTljMWFkNjAwNW11	请求头参数，若开启鉴权，每个接口都会校验该参数。

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

1.1.3 获取所有用户

* 接口描述

请求该接口可以获取到当前设备上保留的所有可登录用户

* 请求地址

/api/systemctrl/users/list

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```

{
  "result": "ok",
  "data": [
    {
      "id": "admin",
      "alias": "Admin",
      "web": true,
      "api": true,
      "create_time": "1970-01-01 00:00:00"
    }
  ]
}

```

* 响应参数说明

字段名	字段类型	字段描述
id	String	用户名
alias	String	别名
web	Boolean	是否允许在 web 端新增用户
api	Boolean	
create_time	String	创建时间

1.1.4 添加用户

* 接口描述

由管理员或者其他已经登录的用户添加新用户的接口

* 请求地址

/api/systemctrl/users/add

* 请求方法

POST

* 请求参数

```

{
  "id": "test_user",
  "password": "qwer1234",
  "confirmNewPassword": "qwer1234",
  "alias": "test_user",
  "enable_web": true,
  "enable_api": true
}

```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
id	String	Y	admin	用户名
password	String	Y	Admin123	密码

confirmNewPassword	String	Y	Admin123	确认密码
alias	String	Y	admin	别名
enable_web	Boolean	Y	true	是否允许在 web 端新增用户
enable_api	Boolean	Y	true	

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

1.1.5 删除用户

* 接口描述

在请求该接口时，传入单个用户名，或者用户名组成的列表，随后即可删除相应的用户。

* 请求地址

/api/systemctrl/users/remove

* 请求方法

POST

* 请求参数

单个删除样例

```
{
  "ids": "Jack"
}
```

批量删除样例

```
{
  "ids": ["Jack", "Bob"]
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
ids	String/Array	Y	["Jack"]	用户名/用户名列表

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

1.1.6 修改用户信息

* 接口描述

请求此接口，可以修改设备上已经保存的之外的用户信息。用户名必填，其他选项均为选填

* 请求地址

/api/systemctrl/users/modify

* 请求方法

POST

* 请求参数

```
{
  "id": "test1",
  "alias": "test1",
  "enable_web": true,
  "enable_api": true,
  "password": "Admin001"
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
id	String	Y	admin	用户名
alias	String	N	user1	别名
enable_web	boolean	N	true	是否允许在 web 端新增用户
enable_api	boolean	N	true	
password	String	N	Admin123	密码

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

1.1.7 获取认证信息

* 接口描述

获取鉴权认证信息

* 请求地址

/api/systemctrl/users/auth_info

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "data": {
    "enable": false
  }
}
```

* 响应参数说明

字段名	字段类型	字段描述
enable	Boolean	是否开启鉴权（默认关闭）

1.1.8 设置认证信息

* 接口描述

开关 api 鉴权

* 请求地址

/api/systemctrl/users/auth_info

* 请求方法

POST

* 请求参数

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

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
enable	Boolean	Y	true	是否开启鉴权

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

1.2 时间，时区设置/Area and time

1.2.1 设置地区时区

* 接口描述

请求此接口用于设置地区时区，需要传入指定地区的标准时字符串

* 请求地址

/api/systemctrl/systime/setLocation

* 请求方法

POST

* 请求参数

```
{
  "timezone": "Asia/Shanghai",
  "offset": 8,
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
timezone	String	Y	"Asia/Shanghai"	时区
offset	Integer	N	8	当 timezone 无法找到时,会使用此字段,此字段为格林威治时间基础上加减的小时,如 GMT+8, 范围为-14 到 12

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

1.2.2 设置系统时间

* 接口描述

请求此接口设置系统时间。可以选择与当前 PC 对时、手动校时，或者从 NTP 服务器同步三种方式设置。

* 请求地址

/api/systemctrl/systime/setTime

* 请求方法

POST

* 请求参数

```
{  
  "time": "2024-10-11 10:29:30",  
  "ntp": "ntp1.aliyun.com ntp2.aliyun.com",  
  "timetype": "pc"  
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
timetype	String	Y	"pc"	时间设置方式 pc : 与当前 PC 对时; manual : 手动校时; ntp : 从 NTP 服务器同步
time	String	N	"2023-08-07 17:37:14"	标准格式的时间, 格式为 "Y-m-d H:M:S"。当 timetype 参数值为 pc 或 manual 时, 必填!
ntp	String	N	"ntp1.aliyun.com ntp2.aliyun.com ntp3.aliyun.com"	ntp 服务器地址, 多个服务器之前用空格间隔。当 timetype 参数值为 ntp 时, 必填!

* 响应参数

```
{  
  "result": "ok",  
  "msg": null  
}
```

* 响应参数说明

无

1.2.3 获取系统时间时区

* 接口描述

获取系统时间时区信息

* 请求地址

/api/systemctrl/systemtime/get

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "msg": null,
  "data": {
    "time": "2024-07-31 18:04:56",
    "timetype": "pc",
    "Timezone": "Asia/Shanghai",
    "offset": 8,
    "ntp": "ntp1.aliyun.com ntp2.aliyun.com ntp3.aliyun.com"
  }
}
```

* 响应参数说明

字段名	字段类型	字段描述
time	String	系统当前时间
timetype	String	时间设置方式
Timezone	String	时区
offset	Integer	当 timezone 无法找到时,会使用此字段,此字段为格林威治时间基础上加减的小时,如 GMT+8, 范围为-14 到 12
ntp	String	ntp 服务器地址, 多个服务器之前用空格间隔。

1.2.4 获取 ntp 状态

* 接口描述

获取当前 ntp 状态, 只有设置了 ntp 时, 此接口才有返回

* 请求地址

/api/systemctrl/systemtime/ntpstatus

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
["string"]
```

* 响应参数说明

返回列表，列表中的值为对应 ntp 状态

1.3 系统设置/System setting

1.3.1 获取开源协议

* 接口描述

请求此接口获取开源协议，固定值。

* 请求地址

/api/systemctrl/system/components

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "data": {
    "components": [
      {
        "name": "vue",
        "url": "MIT",
        "software": "MIT License",
        "address": "https://github.com/vuejs/vue"
      },
      {
        "name": "Element-UI",
        "url": "MIT",
        "software": "MIT License",
        "address": "https://github.com/ElementFE/element"
      },
      {
        "name": "python3",
        "url": "PSF",
        "software": "PSF",
        "address": "http://www.python.org"
      },
      {
        "name": "nginx",
        "url": "BSD2C",
        "software": "BSD-2c",

```

```

    "address": "http://nginx.org"
  },
  {
    "name": "libev",
    "url": "BSD2C",
    "url2": "GPL",
    "software": "BSD-2c",
    "software2": "GPLv2+",
    "address": "http://dist.schmorp.de/libev"
  },
  {
    "name": "openssl",
    "url": "OpenSSL",
    "url2": "SSLey",
    "software": "OpenSSL",
    "software2": "SSLey",
    "address": "http://www.openssl.org"
  }
]
}

```

* 响应参数说明

字段名	字段类型	字段描述
name	String	协议名称
url	String	协议域名
software	String	协议版本
address	String	协议地址

1.3.2 获取设备名称

* 接口描述

请求此接口获取设备的名称，该名称通常由用户指定，没有手动设置的情况下，默认为设备型号+序列号。

* 请求地址

/api/systemctrl/system/getHostname

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "msg": null,
  "data": {
    "hostname": "E3-201412000F1DB"
  }
}
```

* 响应参数说明

字段名	字段类型	字段描述
hostname	String	设备名

1.3.3 设置设备名称

* 接口描述

设置设备名称。

* 请求地址

/api/systemctrl/system/setHostname

* 请求方法

GET

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
name	String	Y	"E3"	设备名称

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

1.3.4 获取 http 和 https 端口开关

* 接口描述

此接口提供给前端，用户获取当前设备上保存的 http 和 https 的端口开关开闭情况，并显示在页面上。

* 请求地址

/api/systemctrl/system/getProtocol

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "msg": null,
  "data": {
    "http": true,
    "https": true,
    "httpPort": 80,
    "httpsPort": 443
  }
}
```

* 响应参数说明

字段名	字段类型	字段描述
http	Boolean	是否开启 http 连接
https	Boolean	是否开启 https 连接
httpPort	Integer	http 端口号
httpsPort	Integer	https 端口号

1.3.5 设置 http 和 https 端口开关

* 接口描述

默认情况下，本机器 https 不开启，如果想要使用语音对讲功能，可以通过此接口打开 https 的开关

* 请求地址

/api/systemctrl/system/setProtocol

* 请求方法

POST

* 请求参数

```
{
  "http": true,
  "https": true,
  "httpPort": 80,
  "httpsPort": 443
}
```

```
}  
// http 和 https 不能同时关闭
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
http	Boolean	Y	true	是否开启 http 连接
https	Boolean	Y	true	是否开启 https 连接
httpPort	Integer	Y	80	设置 http 端口号
httpsPort	Integer	Y	443	设置 https 端口号

* 响应参数

```
{  
  "result": "ok",  
  "msg": null  
}
```

* 响应参数说明

无

1.3.6 获取 ssl 证书和密钥

* 接口描述

获取用户自定义证书和密钥的存放地址，并返回重置证书按钮是否可用。

* 请求地址

/api/systemctrl/system/ssl

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{  
  "result": "ok",  
  "msg": null,  
  "data": {  
    "ssl_crt": "",  
    "ssl_key": "",  
    "enable_reset": false  
  }  
}
```

* 响应参数说明

字段名	字段类型	字段描述
ssl_cert	String	自定义证书存放地址。如果没有上传自定义证书则为空
ssl_key	String	自定义密钥存放地址。如果没有上传自定义密钥则为空
enable_reset	Boolean	重置证书是否可用

1.3.7 上传 ssl 证书和密钥

* 接口描述

用于用户上传自定义的 ssl 证书和密钥。上传后不会自动应用，若上传后未应用再次上传，不管上传的文件名是否相同，都会覆盖之前上传的文件。

* 请求地址

/api/systemctrl/system/ssl/{file_type}

* 请求方法

POST

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
file_type	String	Y	cert	"cert": 证书文件; "key": 密钥文件
file	Binary	Y	文件	证书或密钥文件。证书要求.crt 或.pem, 密钥要求.key

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

1.3.8 应用证书

* 接口描述

应用用户之前上传的证书和密钥。会校证书和密钥的合规性、匹配性。

* 请求地址

/api/systemctrl/system/ssl

* 请求方法

POST

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{  
  "result": "ok",  
  "msg": null  
}
```

* 响应参数说明

无

1.3.9 重置证书

* 接口描述

用于将 ssl 证书和密钥恢复到出厂设置

* 请求地址

/api/systemctrl/system/resetSsl

* 请求方法

POST

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{  
  "result": "ok",  
  "msg": null  
}
```

* 响应参数说明

无

1.3.10 恢复出厂设置

* 接口描述

通过请求该接口，后期所有在当前设备上做的设置将被清除，恢复到当前版本的全新状态。

* 请求地址

/api/systemctrl/system/reFactory

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{  
  "result": "ok",  
  "msg": null  
}
```

* 响应参数说明

无

1.3.11 复位

* 接口描述

通过请求该接口，可以重启 codec 编解码服务。

* 请求地址

/api/systemctrl/system/restore

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{  
  "result": "ok",  
  "msg": null  
}
```

* 响应参数说明

无

1.3.12 重启设备

* 接口描述

通过请求该接口，设备将立即重启

* 请求地址

/api/systemctrl/system/reboot

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{  
  "result": "ok",  
  "msg": null  
}
```

* 响应参数说明

无

1.3.13 ping

* 接口描述

该接口是测试设备与目标 IP 的连通性

* 请求地址

/api/systemctrl/system/ping

* 请求方法

POST

* 请求参数

```
{  
  "target_addr": "192.168.35.22"  
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
target_addr	String	Y	192.168.35.22	目标 IP

* 响应参数

```
{
  "result": "ok",
  "data": {
    "loss_rate": 100,
    "reachable": false
  },
  "msg": null
}
```

* 响应参数说明

字段名	字段类型	字段描述
loss_rate	Integer	丢失率
reachable	Boolean	目标 IP 是否可达

1.3.14 获取网卡速度

* 接口描述

当前设备上有 eth0 和 eth1 两张网卡。在请求时该接口时，可以选择指定网卡返回速度。如果不给参数，则默认将两张网卡的速度都返回给用户。

* 请求地址

/api/systemctrl/system/getNetworkSpeed

* 请求方法

GET

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
iframe	String	N	eth0	当请求给了该参数时，将按照当前参数返回对应的网卡速度。未填写时，eth0 和 eth1 都返回

* 响应参数

```
{
  "result": "ok",
  "data": {
    "eth0": {
      "receive": 71.45,
      "transmit": 7741.28,
      "speed": 1000
    },
    "eth1": {}
  }
}
```

* 响应参数说明

字段名	字段类型	字段描述
receive	Float	两次接口调用间的接收数据量
transmit	Float	发送数据量
speed	Integer	网卡速度

1.3.15 获取系统信息

* 接口描述

通过给定的参数来获取指定的信息，当前版本设备参数可在设备固件版本、cpu 使用率、内存的使用率、磁盘使用率、持续时间中进行选择开启或者关闭

* 请求地址

/api/systemctrl/system/getSystemInfo

* 请求方法

GET

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
version	Boolean	N	true	版本信息,软件版本,硬件版本,NDI 版本,序列号
cpu	Boolean	N	true	返回 cpu 利用率 precent(%)
memory	Boolean	N	true	返回内存使用率 precent(%),内存总量 total(kb),内存使用量 used(kb)
disk	Boolean	N	true	硬盘利用率 precent(%),硬盘总量 total(kb),硬盘剩余 free(kb),硬盘使用量 used(kb)
persisTime	Boolean	N	true	系统运行时间 uptime(s)

* 响应参数

```
{
  "result": "ok",
  "data": {
    "version": {
      "ndiVersion": "5.1.3",
      "hardwareVersion": "3.0",
      "softwareVersion": "1.00.0003",
      "serialNumber": "201412000F1DB",
      "product": "P3",
      "oem": "kiloview"
    },
    "cpu": {
      "precent": 10.891
    }
  }
}
```

```

    },
    "mem": {
      "precent": 27.3,
      "total": 1029500,
      "used": 280996
    },
    "disk": {
      "precent": 27,
      "free": 1368528,
      "used": 508904,
      "total": 1998672
    },
    "persisTime": {
      "uptime": 71664.16
    },
    "temp": {
      "temp": 40
    }
  }
}

```

* 响应参数说明

字段名	字段类型	字段描述
version	Object	版本信息,软件版本,硬件版本,NDI 版本,序列号
cpu	Object	返回 cpu 利用率 precent(%)
mem	Object	返回内存使用率 precent(%),内存总量 total(kb),内存使用量 used(kb)
disk	Object	硬盘利用率 precent(%),硬盘总量 total(kb),硬盘剩余 free(kb),硬盘使用量 used(kb)
persisTime	Object	系统运行时间 uptime(s)

1.3.16 上传升级固件

* 接口描述

该接口用于上传升级固件，该接口在单独使用时需要构造文件参数才可以继续。

* 请求地址

/api/systemctrl/system/upload

* 请求方法

POST

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
file	Binary	Y	bin 文件	固件包文件

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

1.3.17 分片上传升级固件

* 接口描述

文件分片上传

* 请求地址

/api/systemctrl/system/uploadfile

* 请求方法

POST

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
identifier	String	Y	admin	文件唯一标识符
number	String	Y	0	文件分片序号(初始值为0)
file_type	String	Y	bin	文件类型/后缀
file	Binary	Y	bin 文件	固件包

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

1.3.18 固件合并

* 接口描述

固件合并

* 请求地址

/api/systemctrl/system/mergefile

* 请求方法

POST

* 请求参数

```
{  
  "identifier": "string",  
  "number": "string",  
  "file_type": "string"  
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
identifier	String	Y	admin	文件唯一标识符
number	String	Y	2	文件分片序号
file_type	String	Y	bin	文件类型/后缀

* 响应参数

```
{  
  "result": "ok",  
  "msg": null  
}
```

* 响应参数说明

无

1.3.19 产品测试服务开关

* 接口描述

该接口用于提供给 webui-production 页面，开启 py-production-module 的开关

* 请求地址

/api/systemctrl/system/production_off

* 请求方法

POST

* 请求参数

```
{  
  "value": true  
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
------	------	------	-------	----

value	Boolean	Y	true	产品测试服务的开关。此处标识设置开关
-------	---------	---	------	--------------------

* 响应参数

```
{
  "result": "ok",
  "data": {
    "value": true
  }
}
```

* 响应参数说明

字段名	字段类型	字段描述
value	Boolean	产品测试服务的开关。此处反映开关是否激活

1.3.20 服务的启停状态

* 接口描述

用于获取 production 服务的开关状态

* 请求地址

/api/systemctrl/system/production_stat

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "data": {
    "value": true
  }
}
```

* 响应参数说明

字段名	字段类型	字段描述
value	Boolean	产品测试服务模块的状态。返回当前服务是否有激活。

1.4 系统预设/System preset

1.4.1 获取模块列表

* 接口描述

请求此接口获取所有模块列表

* 请求地址

/api/systemctrl/system_preset/modules

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "data": [
    "time",
    "user",
    "ptz",
    "edid",
    "maintenance",
    "codec_stream",
    "record",
    "talk",
    "osd",
    "panel"
  ]
}
```

* 响应参数说明

字段名	字段类型	字段描述
data	Array	模块列表。time: 区域和时间设置; user: 用户管理设置; ptz: PTZ 串口设置; edid: EDID 设置; timed_reboot: 定时重启设置; codec_stream: 编码及流参数设置; record: 录像配置; talk: 语音通话; osd: osd 图文叠加; panel: 小屏面板设置

1.4.2 获取预设配置

* 接口描述

请求此接口获取预设设置列表

* 请求地址

/api/systemctrl/system_preset/get

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "data": [
    {
      "pid": "df5bbf9a-3c14-4444-a477-ab77e407cb7f",
      "name": "888"
    },
    {
      "pid": "b68bbc3b-a35d-45a7-b552-6b3c264b0797",
      "name": "777"
    }
  ]
}
```

* 响应参数说明

字段名	字段类型	字段描述
pid	String	预设 ID, 唯一且不重复
name	String	预设名称, 唯一且不重复

1.4.3 导入预设配置到此设备

* 接口描述

请求此接口导入预设配置到本设备

* 请求地址

/api/systemctrl/system_preset/import

* 请求方法

POST

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
------	------	------	-------	----

file	Binary	Y	预设配置 文件	预设文件，可以从其他设备中导出预设配置 文件，导入该设备当中
------	--------	---	------------	-----------------------------------

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

1.4.4 保存模块配置到预设

* 接口描述

请求此接口保存系统配置到预设

* 请求地址

/api/systemctrl/system_preset/save

* 请求方法

POST

* 请求参数

```
{
  "name": "preset_name",
  "modules": ["time"]
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
name	String	Y	admin	预设名称，唯一且不重复
modules	Array	Y	["time"]	模块名称，通过"获取模块列表"获取，表示需要保存哪些配置文件到预设配置中

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

1.4.5 下载模块配置到本机

* 接口描述

请求此接口保存系统配置到预设

* 请求地址

/api/systemctrl/system_preset/download

* 请求方法

GET

* 请求参数

name=123&modules=["time"]

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
name	String	Y	admin	预设文件名称
modules	String	Y	["time","osd"]	模块名称，通过"获取模块列表"获取，表示需要保存哪些配置文件到预设配置中

* 响应参数

```
file: File(...)
content-disposition: attachment; filename={filename}.zip
content-type: application/octet-stream
```

* 响应参数说明

下载文件

1.4.6 载入预设配置到设备

* 接口描述

请求此接口载入预设配置到此设备

* 请求地址

/api/systemctrl/system_preset/{pid}/load

* 请求方法

POST

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
pid	String	Y	admin	预设 ID

* 响应参数

```
{
  "result": "ok",
```

```
"msg": null
}
```

* 响应参数说明

无

1.4.7 删除预设配置

* 接口描述

请求此接口删除预设配置

* 请求地址

/api/systemctrl/system_preset/{pid}

* 请求方法

DELETE

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
pid	String	Y	admin	预设 ID

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

1.4.8 导出预设配置到本机

* 接口描述

请求此接口导出预设配置到本机

* 请求地址

/api/systemctrl/system_preset/{pid}/export

* 请求方法

GET

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
pid	String	Y	admin	预设 ID

* 响应参数

无

* 响应参数说明

下载文件

1.5 日志服务/System log

1.5.1 获取日志

* 接口描述

获取对应模块的日志

* 请求地址

/api/systemctrl/system_log/get/{logname}

* 请求方法

GET

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
logname	String	Y	codec	模块名称，具体为： codec/streamer/record/network/system，分别表示编码服务日志、流媒体服务日志、录像服务日志、网络状态日志、系统服务日志

* 响应参数

```
{
  "result": "ok",
  "data": "[2024-07-18 17:24:23,437] INFO:    <codec> /tmp/video_main,
status=<enum GST_STATE_CHANGE_ASYNC of type Gst.StateChangeReturn>\n[2024-
07-18 17:24:23,451] INFO:    <codec> /tmp/video_sub, status=<enum
GST_STATE_CHANGE_ASYNC of type Gst.StateChangeReturn>\n[2024-07-18
17:24:23,460] INFO:    <codec> /tmp/video_preview, status=<enum
GST_STATE_CHANGE_ASYNC of type Gst.StateChangeReturn>\n[2024-07-18
17:24:23,718]"
}
```

* 响应参数说明

字段名	字段类型	字段描述
result	String	"ok"表示成功, "error"表示错误
data	String	日志内容

1.5.2 导出系统日志

* 接口描述

导出系统日志

* 请求地址

/api/systemctrl/system_log/export/{time}

* 请求方法

GET

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
time	String	Y	old	old 或 new, 分别表示系统历史日志和系统当前日志

* 响应参数

```
file: File(...)
content-disposition: attachment; filename={filename}.zip
content-type: application/octet-stream
```

* 响应参数说明

下载日志文件

1.6 系统维护/System Maintenance

1.6.1 获取系统维护配置

* 接口描述

获取系统维护配置

* 请求地址

/api/systemctrl/system_maintenance/get

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "data": {
    "enable": false,
    "policy": "day",
    "start_date": "1970-01-01 00:00:00",
    "months": [],
    "weeks": [],
    "days": [],
    "day": 1
  }
}
```

* 响应参数说明

字段名	字段类型	字段描述
enable	Boolean	系统维护启停状态
policy	String	系统维护启用方式, 仅可为 day、week、month, 分别表示按日、按周、按月策略
start_date	String	开始时间, 可以设置为未来的一个时间, 定时重启任务将从设定时间开始执行, 时间格式必须是"%Y-%m-%d %H:%M:%S"的时间字符串
months	Array	月配置, 数据必须是[1, 12]的区间内
weeks	Array	周配置, 数据必须是[0, 6]的区间内, 0 表示周日, 1 表示周一, 2 表示周二, 以此类推
days	Array	天配置, 数据必须是[1, 31]的区间内
day	Integer	数据必须是[1, 31]的区间内, 每隔多少天进行重启

1.6.2 修改系统维护配置

* 接口描述

修改系统维护配置

* 请求地址

/api/systemctrl/system_maintenance/set

* 请求方法

POST

* 请求参数

```
{
  "enable": false,
  "policy": "day",
  "start_date": "1970-01-01 00:00:00",
  "months": [
    12
  ],
  "weeks": [
    6
  ],
  "days": [
    31
  ],
  "day": 31
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
enable	Boolean	Y	false	系统维护启停状态
policy	String	Y	day	系统维护启用方式
start_date	String	Y	1970-01-01 00:00:00	开始时间
months	Array	Y	[1,2]	月配置
weeks	Array	Y	[6,0]	周配置
days	Array	Y	[1,5]	天配置
day	Integer	Y	1	具体天

* 响应参数

```
{
  "result": "ok",
}
```

* 响应参数说明

无

2 录制服务/Service of record

2.1 录制配置/Record

2.1.1 删除录制文件

* 接口描述

删除录制文件

* 请求地址

/api/record/record/remove_file

* 请求方法

POST

* 请求参数

```
{  
  "path": "/run/media/sda1/RECORD/HDMI20231106_171002_497891/REC-HDMI-  
202311061710024978911.mp4"  
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
path	String	Y	/run/media/sda1/RECORD/HDMI20231106_171002_497891/REC-HDMI-202311061710024978911.mp4	待删除的录制文件路径

* 响应参数

```
{  
  "result": "ok",  
  "msg": null  
}
```

* 响应参数说明

无

2.1.2 获取录制文件列表

* 接口描述

获取录制文件列表

* 请求地址

/api/record/record/get_disk_files

* 请求方法

GET

* 请求参数

device=/run/media/sda1

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
device	String	Y	/run/media/sda1	挂载设备路径
stream	String	N	hdmi	流类别

* 响应参数

```
{
  "result": "ok",
  "data": [
    {
      "path": "/run/media/sda1/RECORD/HDMI20231106_174429_641406/REC-
HDMI-2023110617442964140636.mp4",
      "name": "REC-HDMI-2023110617442964140636.mp4",
      "size": 12.84,
      "current": true,
      "stream": "hdmi",
      "start_time": "2023-11-06 17:44:29",
      "end_time": null
    },
    {
      "path": "/run/media/sda1/RECORD/HDMI20231106_174330_644981/REC-
HDMI-2023110617433064498135.mp4",
      "name": "REC-HDMI-2023110617433064498135.mp4",
      "size": 15.06,
      "current": false,
      "stream": "hdmi",
      "start_time": "2023-11-06 17:43:30",
      "end_time": "2023-11-06 17:44:29"
    }
  ]
}
```

* 响应参数说明

字段名	字段类型	字段描述
path	String	录制文件路径
name	String	录制文件名称
size	Integer	文件大小，单位为 MB
current	Boolean	是否为当前录制文件，如果是正在录制文件则不允许下载，否则播放时会出现文件损坏的标识
stream	String	视频输入源，hdmi/sdi/pip
start_time	String	录制开始时间
end_time	String	录制结束时间

2.1.3 勾选 usb 网盘

* 接口描述

通过本接口可以勾选 usb 盘，表示录制视频时将会录制到该 usb 盘。

* 请求地址

/api/record/record/choose_usb

* 请求方法

POST

* 请求参数

```
{
  "id": "123",
  "choose": true
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
id	String	Y	admin	存储设备的 ID，nas 盘为用户设置的唯一 ID，usb 盘为自动生成的 sda、sdb 等盘符
choose	Boolean	Y	true	是否选择作为录制存储盘，不勾选则录制时不往该存储盘录制，反之亦然

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

2.1.4 勾选 nas 网盘

* 接口描述

通过本接口可以勾选 nas 盘，表示录制视频时将会录制到该 nas 盘。

* 请求地址

/api/record/record/choose_nas

* 请求方法

POST

* 请求参数

```
{
  "id": "123",
  "choose": true
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
id	String	Y	admin	存储设备的 ID, nas 盘为用户设置的唯一 ID, usb 盘为自动生成的 sda、sdb 等盘符
choose	Boolean	Y	true	是否选择作为录制存储盘, 不勾选则录制时不往该存储盘录制, 反之亦然

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

2.1.5 存储列表

* 接口描述

获取存储介质列表

* 请求地址

/api/record/record/disk_list

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "data": {
    "nas": [
      {
        "id": "123",
        "name": "123",
        "type": "cifs",
        "host": "192.168.43.1",
        "volume": "abc",
        "options": "user=,password=",
        "mounted": "/media/nas/123",
        "filesystem": "",
        "status": "mounted",
        "choose": false,
        "disk_type": ""
      }
    ]
  }
}
```

```

        "total": 283584,
        "used": 41314,
        "free": 242270,
        "percent": 14
    }
],
"usb": [
    {
        "id": "sda1",
        "name": "sda1",
        "type": "",
        "host": "",
        "volume": "/dev/sda1",
        "options": "",
        "mounted": "/run/media/sda1",
        "filesystem": "vfat",
        "status": "mounted",
        "choose": true,
        "disk_type": "usb",
        "total": 29534,
        "used": 23948,
        "free": 5585,
        "percent": 81
    }
]
}
}
}

```

* 响应参数说明

字段名	字段类型	字段描述
id	String	用户指定的唯一 ID
name	String	用户指定的 nas 盘名称
type	String	仅 nas 盘存在类型, "cifs", "nfs"
host	String	仅 nas 盘存在主机名, 文件系统主机名
volume	String	盘符路径
options	String	挂载选项
mounted	String	挂载的 Unix 路径地址
filesystem	String	仅 u 盘存在文件系统类型, 文件系统类型, vfat、ntfs
status	String	挂载状态, mounted (已挂载) /mounting (挂载中) /mount_fail (挂载失败)
choose	Boolean	是否选中
disk_type	String	磁盘类型, 无太多意义
total	Integer	当前磁盘总大小, 单位为 MB
used	Integer	当前磁盘已使用大小, 单位为 MB

free	Integer	当前磁盘剩余大小，单位为 MB
percent	Integer	当前磁盘使用率，百分比，0-100

2.1.6 录制信息

* 接口描述

通过本接口可以获取用户设置的录制信息。

* 请求地址

/api/record/record/get_record_info

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "data": {
    "disk_type": "largest",
    "format": "mp4",
    "file_prefix": "REC",
    "limit_type": "size-loop",
    "limit_size": 512,
    "limit_time": 3600,
    "disk_policy": "overwrite",
    "follow_timecode_start": false
  }
}
```

* 响应参数说明

字段名	字段类型	字段描述
disk_type	String	指定分区，暂时没有使用，无实际意义，可忽略
format	String	录制格式，mp4、mov
file_prefix	String	文件前缀，0-10 位字符串即可
limit_type	String	文件限制类型，none 为不限制大小与时间，time-loop 为按时长分割，size-loop 为按大小尺寸分割
limit_size	Integer	文件限制尺寸，当 limit_type 为 size-loop 将会被使用，大小范围为 512-51200MB
limit_time	Integer	文件限制时长，当 limit_type 为 time-loop 将会被使

		用，时间范围为 60-14400s
disk_policy	String	磁盘策略， overwrite （磁盘空间不足时，将自动覆盖老的录制文件）， noSpace （磁盘空间不足时）
follow_timecode_start	Boolean	是否跟随 sdi 时间码启动, 为 true 且 sdi 开启且连接 sdi 摄像头时，sdi 时间码产生变化则直接启停录制

2.1.7 视频录制配置

* 接口描述

通过本接口可以自行设置录制信息

* 请求地址

/api/record/record/settings

* 请求方法

POST

* 请求参数

```
{
  "format": "mp4",
  "file_prefix": "REC",
  "limit_type": "time-loop",
  "limit_size": 51200,
  "limit_time": 14400,
  "disk_policy": "noSpace",
  "follow_timecode_start": false
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
format	String	Y	MP4	录制格式，mp4、mov、ts
file_prefix	String	Y	REC	文件前缀，0-10 位字符串即可
limit_type	String	Y	time_loop	文件限制类型，none 为不限制大小与时间，time-loop 为按时长分割，size-loop 为按大小尺寸分割
limit_size	Integer	Y	512	文件限制尺寸，当 limit_type 为 size-loop 将会被使用，大小范围为 512-51200MB
limit_time	Integer	Y	60	文件限制时长，当 limit_type 为 time-loop 将会被使用，时间范围为 60-14400s

disk_policy	String	Y	noSpace	磁盘策略，overwrite（磁盘空间不足时，将自动覆盖老的录制文件），noSpace（磁盘空间不足时）
follow_timecode_start	Boolean	N	false	是否跟随 sdi 时间码启动, 为 true 且 sdi 开启且连接 sdi 摄像头时, sdi 时间码产生变化则直接启停录制

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

2.2 录制按钮/Recording

2.2.1 视频录制

* 接口描述

通过本接口可以开启、关闭录制模块

* 请求地址

/api/record/{interface}/recording

* 请求方法

POST

* 请求参数

```
{
  "start": true
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
start	Boolean	Y	true	是否开启录制，为 true 则开启录制，为 false 则关闭录制

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

2.2.2 获取录制状态

* 接口描述

通过本接口可以查看录制状态

* 请求地址

/api/record/{interface}/get_recording_status

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "data": {
    "status": false
  }
}
```

* 响应参数说明

字段名	字段类型	字段描述
result	String	"ok"表示成功, "error"表示错误
status	Boolean	是否开启

2.3 网络存储/NAS Manager

2.3.1 添加网络存储

* 接口描述

通过本接口可以给当前设备手动添加网络存储（cifs、nfs 文件系统）

* 请求地址

/api/record/nas/add

* 请求方法

POST

* 请求参数

```
{
  "id": "123",
  "name": "123",
  "type": "cifs",
  "host": "192.168.43.1",
  "volume": "abc",
  "options": "user=,password="
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
id	String	Y	123	用户指定的唯一 ID
name	String	Y	123	用户指定的 nas 盘名称
type	String	Y	cifs	类型: "cifs", "nfs"
host	String	Y	192.168.35.22	nas 文件系统 host
volume	String	Y	abc	挂载目录
options	String	Y	user=,password=	挂载参数

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

2.3.2 移除网络存储

* 接口描述

通过本接口可以移除设备中的网络存储（cifs、nfs 文件系统）

* 请求地址

/api/record/nas/remove

* 请求方法

POST

* 请求参数

```
{
  "id": "123"
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
id	String	Y	123	用户指定的唯一 ID

* 响应参数

```
{  
  "result": "ok",  
  "msg": null  
}
```

* 响应参数说明

无

3 流服务/Service of stream

3.1 流服务/Stream

3.1.1 获取流列表

* 接口描述

获取推流列表信息

* 请求地址

/api/streamer/{interface}/stream/list

* 请求方法

GET

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
interface	String	Y	hdmi	hdmi: HDMI 接口, sdi: SDI 接口, pip: 画中画, uvc: usb Camera

* 响应参数

```
[  
  {  
    "id": "a08290a345356a77",  
    "type": "ndi_hx",  
    "name": "NDI-HX",  
    "enable": false,  
    "status": "stop",  
    "bindAudio": 1,  
    "ttl": 1,  
    "channel_name": "channel-hdmi",  
    "group": "",  
    "netmask": "255.255.0.0",  
    "connection": "disable_rudp",  
    "netprefix": "239.255.0.0",  
    "discovery_server": "",  
    "addressUrl": "offline"  
  }  
]
```

```
}  
]
```

* 响应参数说明

字段名	字段类型	字段描述
-	Array	每个 OBJECT 存储的都是流的状态，对应流包含什么属性可以参考创建流时的参数

3.1.2 添加流

* 接口描述

添加一路推流

* 请求地址

/api/streamer/{interface}/stream

* 请求方法

POST

* 请求参数

```
{  
  "port": 554,  
  "httpTunnelPort": 8554,  
  "session": "ch01",  
  "auth": true,  
  "multicast_enable": false,  
  "multicast_addr": "224.0.0.1",  
  "multicast_ttl": 127,  
  "multicast_port_min": 31000,  
  "multicast_port_max": 31004,  
  "name": "test",  
  "type": "rtsp",  
  "bindAudio": 1,  
  "bindVideo": "main"  
}
```

* RTSP 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
interface	String	Y	hdmi	路径参数，将拼接在 URL 中。 hdmi/sdi/pip/uvc
port	Integer	Y	554	RTSP 占用端口
httpTunnelPort	Integer	Y	8554	Tunnel 端口
session	String	Y	ch01	会话 ID，将作为推流地址的路径参数

auth	Boolean	Y	true	是否开启用户验证
multicast_enable	Boolean	Y	true	是否开启组播
multicast_addr	String	Y	224.0.0.1	组播地址
multicast_ttl	Integer	Y	127	组播 TTL 值
multicast_port_min	Integer	Y	31000	组播最小端口号
multicast_port_max	Integer	Y	31004	组播最大端口号
name	String	Y	test	流名称
type	String	Y	rtsp	流类型，默认为 rtsp
bindAudio	Integer	Y	1	音频编码 ID，决定用户使用哪个音频通道

* **RTMP/RTMPS 请求参数说明**

参数名称	参数类型	是否必填	参数值样例	描述
interface	String	Y	hdmi	路径参数，将拼接在 URL 中。 hdmi/sdi/pip/uvc
address	String	Y	rtmp://youtube.com/live2/test	rtmp 推流地址
bindAudio	Integer	Y	1	音频编码 ID，决定用户使用哪个音频通道
bindVideo	String	N	main	使用的视频编码配置是主码流还是子码流。不传默认 main
connIntv	Integer	Y	3	重连间隔，单位为秒，最多重连十次
connTimeout	Integer	Y	15	连接超时，单位为秒，指的是 socket 的 timeout
name	String	Y	123	流名
user	String	Y	user	RTMP 服务器认证账号，可在媒体用户管理设置
password	String	Y	user	RTMP 服务器认证密码
oldRtmp	Boolean	N	false	为 True 则使用旧的 rtmp 版本。默认为 False
type	String	Y	rtmp	流类型，默认为 rtmp

* SRT 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
interface	String	Y	hdmi	路径参数，将拼接在 URL 中。hdmi/sdi/pip/uvc
address	String	Y	srt://192.168.40.2	caller/Rendezvous 模式下必填，listener 模式下可传空字符串
bindAudio	Integer	Y	1	音频编码 ID，决定用户使用哪个音频通道
bindVideo	String	N	main	使用的视频编码配置是主码流还是子码流。不传默认 main
connectionMode	String	Y	Listener	SRT 的连接模式。Listener/Caller/Rendezvous
encryption	String	Y	0	加密方式。0：不加密，AES-128：AES128 加密，AES-192：AES192 加密，AES-256：AES256 加密。
latency	Integer	Y	125	传输延时，单位毫秒
listenerPort	Integer	Y	1025	监听端口，Listener/Rendezvous 模式下将会占用端口
name	String	Y	123	流名
passphrase	String	Y	e10adc3949ba	AES 密钥，只有在需要加密时必传，encryption 为 0 时（即不加密模式）可以传空字符串。限制为 10-24 位长度的字符串
srt_stream_id	String	Y		SRT 的流 ID，可以传空字符串
ts_null_multiple	Float	N	0	TS 包的空包倍率，按目前设定码率设置空包倍率，一般为 0、1.2、1.5、2.0、3.0、4.0、5.0
ts_service_name	String	N	Encoder	service name
ts_service_provider	String	N	Encoder device	service provider

ts_transport_stream_id	Integer	N	101	传输 id, Stream ID
ts_tables_version	Integer	N	6	版本号
ts_pcr_period	Integer	N	20	写入 pcr 的间隔, pcr 周期
ts_start_pid	Integer	N	480	start es pid, Stream start PID
ts_pmt_start_pid	Integer	N	481	启动 pmt pid, PMT start PID
type	String	Y	srt	流类型, 默认为 srt

* **HLS 请求参数说明**

参数名称	参数类型	是否必填	参数值样例	描述
interface	String	Y	hdmi	路径参数, 将拼接在 URL 中。 hdmi/sdi/pip/uvc
session	String	Y	ch01	会话 ID, 将作为推流地址的路径参数
bindAudio	Integer	Y	1	音频编码 ID, 决定用户使用哪个音频通道
bindVideo	String	N	main	使用的视频编码配置是主码流还是子码流。不传默认 main
name	String	Y	123	流名
segmentTime	Integer	Y	5	每个 ts 切片长度, 默认为秒 (s)
type	String	Y	hls	流类型, 默认为 hls

* **ts-udp 请求参数说明**

参数名称	参数类型	是否必填	参数值样例	描述
interface	String	Y	hdmi	路径参数, 将拼接在 URL 中。 hdmi/sdi/pip/uvc
address	String	Y	192.168.40.2	对应目标接收地址, 也可以是组播地址
bindAudio	Integer	Y	1	音频编码 ID, 决定用户使用哪个音频通道
bindVideo	String	N	main	使用的视频编码配置是主码流还是子码流。不传默认 main
name	String	Y	123	流名

port	Integer	Y	5700	对应目标接收端口
ttl	Integer	Y	127	组播 ttl
ts_null_multiple	Float	Y	0	TS 包的空包倍率，按目前设定码率设置空包倍率，一般为 0、1.2、1.5、2.0、3.0、4.0、5.0
bindPort	int	N	0	绑定本地 udp 套接字的端口。范围：0 - 65535。默认为 0，表示系统自动选择一个没有使用的端口
type	String	Y	ts	流类型，默认为 ts
ts_service_name	String	Y	Encoder	service name
ts_service_provider	String	Y	Encoder device	service provider
ts_transport_stream_id	Integer	Y	101	传输 id
ts_tables_version	Integer	Y	6	版本号
ts_pcr_period	Integer	Y	20	写入 pcr 的间隔
ts_start_pid	Integer	Y	480	start es pid
ts_pmt_start_pid	Integer	Y	481	启动 pmt pid
ts_pmt_start_pid	Integer	N	55647	绑定本地 udp 套接字的端口。范围：0 - 65535。默认为 0，表示系统自动选择一个没有使用的端口

* RTP 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
interface	String	Y	hdmi	路径参数，将拼接在 URL 中。 hdmi/sdi/pip/uvc
bindAudio	Integer	Y	1	音频编码 ID，决定用户使用哪个音频通道
bindVideo	String	N	main	使用的视频编码配置是主码流还是子码流。不传默认 main
name	String	Y	123	流名
address	String	Y	192.168.40.2	对应目标接收地址
port	Integer	Y	5700	对应目标接收端口

load_type	String	Y	ts	载荷类型。目前只有 ts
ts_service_name	String	Y	Encoder	ts 流高级参数，用于 MPEG-TS 标识服务名称。默认为 "Encoder"
ts_service_provider	String	Y	Encoder device	ts 流高级参数，默认为 "Encoder device"
ts_transport_stream_id	Integer	N	101	传输 id
ts_tables_version	Integer	N	6	版本号
ts_pcr_period	Integer	N	20	写入 pcr 的间隔
ts_start_pid	Integer	N	480	start es pid
ts_pmt_start_pid	Integer	N	481	启动 pmt pid
ts_null_multiple	Float	N	0	TS 包的空包倍率，按目前设定码率设置空包倍率，一般为 0、1.2、1.5、2.0、3.0、4.0、5.0
ttl	Integer	N	127	组播 ttl
type	String	Y	rtp	流类型，默认为 rtp

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

3.1.3 修改流

* 接口描述

修改单个推流配置

* 请求地址

/api/streamer/{interface}/stream/{stream_id}

* 请求方法

PUT

* 请求参数

```
{
  "group": "",
  "channel_name": "channel-hdmi",
  "connection": "disable_rudp",
  "netprefix": "239.255.0.0",
}
```

```

"netmask": "255.255.0.0",
"ttl": 1,
"discovery_server": "",
"id": "a08290a345356a77",
"name": "NDI-HX",
"type": "ndi_hx",
"bindAudio": 1,
"bindVideo": "main"
}

```

* 请求参数说明

下为 NDI 参数说明，其他流参数可参考创建流时的参数。

参数名称	参数类型	是否必填	参数值样例	描述
interface	String	Y	hdmi	路径参数，将拼接在 URL 中。 hdmi/sdi/pip/uvc
id	String	Y	a08290a345356a77	需要修改的流 ID
name	String	Y	test	流名
group	String	Y	public	可空，为空则默认为 public
channel_name	String	Y	channel-hdmi	通道名称
connection	String	Y	disable_rudp	连接方式，默认为 disable_rudp。disable_rudp: 多 tcp, rudp: 可靠的 udp, udp: udp, multicast: 组播
netprefix	String	Y	239.255.0.0	组播地址，连接方式非组播状态下不生效
netmask	String	Y	255.255.0.0	组播子网掩码，连接方式非组播状态下不生效
ttl	Integer	Y	127	组播 TTL，连接方式非组播状态下不生效
discovery_server	String	Y	192.168.40.2	发现服务器
type	String	Y	ndi_hx	流类型，默认为 ndi_hx
bindAudio	Integer	Y	1	音频编码 ID，决定用户使用哪个音频通道

* 响应参数

```

{
  "result": "ok",
  "msg": null
}

```

* 响应参数说明

无

3.1.4 删除流

* 接口描述

根据流 id 删除指定流

* 请求地址

/api/streamer/{interface}/stream/{stream_id}

* 请求方法

DELETE

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
interface	String	Y	hdmi	路径参数，将拼接在 URL 中。 hdmi/sdi/pip/uvc
stream_id	String	Y	a08290a345356a77	流 ID

* 响应参数

```
{  
  "result": "ok",  
  "msg": null  
}
```

* 响应参数说明

无

3.1.5 流开关

* 接口描述

根据流 id 开启/关闭指定流

* 请求地址

/api/streamer/{interface}/stream/enable

* 请求方法

POST

* 请求参数

```
{  
  "id": "a08290a345356a77",  
}
```

```
"enable": true
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
interface	String	Y	hdmi	路径参数，将拼接在 URL 中。 hdmi/sdi/pip/uvc
id	String	Y	a08290a345356a77	流 ID
enable	String	Y	true	流开关

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

3.1.6 复位 NDI

* 接口描述

将 ndi 复位

* 请求地址

/api/streamer/{interface}/stream/ndi/reset

* 请求方法

POST

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

3.1.7 获取 Webrtc

* 接口描述

获取 Webrtc Port 和 Sdp Mode

* 请求地址

/api/streamer/{interface}/stream/get_webrtc

* 请求方法

GET

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
mode	Integer	Y	0	0: 视频; 1: 图片

* 响应参数

```
{  
  "result": "ok",  
  "msg": null  
}
```

* 响应参数说明

无

3.1.8 设置 Webrtc

* 接口描述

设置 Webrct 模式

* 请求地址

/api/streamer/{interface}/stream/set_webrtc

* 请求方法

GET

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
stream_id	String	Y	admin	流 ID
mode	Integer	Y	0	0: 视频; 1: 图片

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

3.1.9 获取媒体用户

* 接口描述

获取 rtsp 服务的媒体用户

* 请求地址

/api/streamer/{interface}/stream/media_user

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "user_list": [
    "admin"
  ]
}
```

* 响应参数说明

字段名	字段类型	字段描述
user_list	Array	媒体用户名列表

3.1.10 添加媒体用户

* 接口描述

添加 rtsp 服务的媒体用户

* 请求地址

/api/streamer/{interface}/stream/media_user

* 请求方法

POST

* 请求参数

```
{
  "username": "admin",
  "password": "Admin123"
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
username	String	Y	admin	用户名
password	String	Y	Admin123	密码

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

3.1.11 删除媒体用户

* 接口描述

根据用户名删除媒体用户

* 请求地址

/api/streamer/{interface}/stream/media_user/{username}

* 请求方法

DELETE

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
username	String	Y	admin	用户名

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

3.2 PTZ 设置/PTZ setting

3.2.1 PTZ 控制

* 接口描述

通过请求本接口，传入相应的动作和速度等，模拟页面上通过面板操控相机的方式，可以直接控制 PTZ 的方向、对焦和缩放。

* 请求地址

/api/streamer/ptz/control

* 请求方法

GET

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
action	String	Y	up	从下列字符串中选则填写: "up"、"left"、"down"、"right"
speed	Integer	N	50	表示要传入的速度，填入数字 0~100，默认 50，像 home、stop-move、stop-all 等等这种可以不用传该值
id	Integer	N	2	表示传入的预设位，主要用于带 preset 结尾的指令，传入数字即可，建议传 1-16

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

3.2.2 获取 PTZ 信息

* 接口描述

该接口主要是获取当前设备上对于 PTZ 相机的连接和控制的配置信息。

* 请求地址

/api/streamer/ptz/info

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "data": {
    "enable": 0,
    "typ": "serial",
    "network": {
      "protocol": "TCP",
      "addr": "",
      "port": 1,
      "ptz_protocol": "Sony Visca",
      "ptz_addr": "1"
    },
    "serial": {
      "device": "none",
      "baudrate": 9600,
      "parity": "none",
      "startBits": 8,
      "endBits": 1,
      "xonxoff": 0,
      "rtscts": 0,
      "ptz_protocol": "Sony Visca",
      "ptz_addr": "1"
    }
  }
}
```

* 响应参数说明

字段名	字段类型	字段描述
enable	Boolean	PTZ 功能是否开启
typ	String	连接方式 network: 网络连接 serial: 串口连接
network	Object	网络连接 protocol: 网络协议。TCP/UDP; addr: PTZ 相机 IP 地址; port: PTZ 相机端口; ptz_protocol: PTZ 协议。"Sony ; Visca"、"Pelco-d"和"Pelco-p"
serial	Object	串口连接 device: 串口驱动; baudrate: 波特率; parity: 校验; startBits: 数据位; xonxoff: XON/XOFF 开关; rtscts: RTS/CTS 开关

3.2.3 获取 PTZ 协议

* 接口描述

当前版本下, 所支持的协议包括 Sony Visca、Pelco-d、Pelco-p, 通过请求该接口可以通过返回字段的 data 字段获取到相应的协议名称。

* 请求地址

/api/streamer/ptz/protocol

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "msg": null,
  "data": [
    "Sony Visca",
    "Pelco-d",
    "Pelco-p"
  ]
}
```

* 响应参数说明

无

3.2.4 获取串口设备

* 接口描述

当有串口设备接入, 或者直接使用串口线插入本机器, 则可以通过此接口, 实现获取到串口设备的设备节点功能

* 请求地址

/api/streamer/ptz/serial_device

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "data": [
    {
      "device": "ttyUSB0",
      "driver": "FTDI"
    },
  ]
}
```

* 响应参数说明

字段名	字段类型	字段描述
device	String	串口设备名称
driver	String	驱动名称

3.2.5 修改 PTZ 信息

* 接口描述

通过本接口，传入相应的字段参数，修改设备上保存的 PTZ 连接和控制信息

* 请求地址

/api/streamer/ptz/modify

* 请求方法

POST

* 请求参数

```
{
  "enable": 1,
  "typ": "network",
  "device": "none",
  "baudrate": 9600,
  "parity": "none",
  "startBits": 8,
  "endBits": 1,
  "xonxoff": 0,
  "rtscts": 0,
  "ptz_protocol": "Sony Visca",
  "ptz_addr": 1,
  "protocol": "UDP",
  "addr": "192.168.38.18",
  "port": 50381
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
enable	Integer	Y	0	ptz 控制开关。0 关闭；1 开启
typ	String	Y	network	ptz 类型。network/serial/usb

device	String	N	ttyUSB0	串口驱动设备
baudrate	Integer	N	9600	波特率
parity	String	N		奇偶校验。
startBits	Integer	N	8	数据位
endBits	Integer	N	1	停止位
xonxoff	Integer	N	0	XON/XOFF 开关
rtscts	Integer	N	0	RTS/CTS 开关
ptz_protocol	String	N	Sony Visca	PTZ 协议，目前可以选填的协议包括如下： "Sony Visca", "Pelco-d", "Pelco-p"
ptz_addr	Integer	N	1	串口地址位置。协议内部的固定参数。数字，1 或者 0
protocol	String	N	UDP/TCP	网络协议
addr	String	N	192.168.35.35	ptz 相机 ip 地址
port	Integer	N	60381	ptz 相机端口

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

3.3 小屏推流/Panel Stream

3.3.1 获取快捷推流状态

* 接口描述

获取快捷推流状态

* 请求地址

/api/streamer/panel_stream/shortcuts_stream

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "shortcutsRecordList": [],
  "shortcutsRecord": false,
  "shortcutsStream": false,
  "shortcutsEnable": false
}
```

* 响应参数说明

字段名	字段类型	字段描述
shortcutsRecordList	Array	加入到快捷推流的流列表
shortcutsRecord	Boolean	快捷录制开关
shortcutsStream	Boolean	快捷推流开关
shortcutsEnable	Boolean	快捷操作状态

3.3.2 快捷推流开关

* 接口描述

快捷推流开关

* 请求地址

/api/streamer/panel_stream/shortcuts_stream

* 请求方法

POST

* 请求参数

```
{
  "enable": true
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
enable	String	Y	true	快捷推流开关

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

3.3.3 设置快捷推流开关状态

* 接口描述

设置快捷推流开关状态。调用此接口仅更改配置文件中快捷推流的状态，不实际开关快捷推流。

* 请求地址

/api/streamer/panel_stream/set_shortcuts_enable

* 请求方法

POST

* 请求参数

```
{
  "enable": true
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
enable	String	Y	true	快捷推流开关

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

3.3.4 获取 tally 灯设置

* 接口描述

获取当前 tally 灯显示绑定的编码(hdmi/sdi/mix)

* 请求地址

/api/streamer/panel_stream/get_tally_setting

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
"sdi"
```

* 响应参数说明

字段名	字段类型	字段描述
"sdi"	String	hdmi/sdi/pip/off 当前 tally 灯显示绑定的编码

3.3.5 更改 tally 灯设置

* 接口描述

更改当前 tally 灯显示绑定的编码(hdmi/sdi/mix)，此接口不设置 tally 灯状态。只有设置为指定编码且该编码的 NDI|HX 流打开时，对该 NDI|HX 流设置 tally 状态，tally 灯才会生效。

* 请求地址

/api/streamer/panel_stream/set_tally_setting

* 请求方法

POST

* 请求参数

```
{
  "param": "hdmi"
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
param	String	Y	"hdmi"	hdmi/sdi/pip/off 当前 tally 灯显示绑定的编码

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

3.4 快捷推流/shortcuts

3.4.1 获取快捷推流列表

* 接口描述

获取当前的快捷推流设置

* 请求地址

/api/streamer/shortcuts/list

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "panel_enable": true,
  "shortcuts_record": [],
  "hdmi": [
    {
      "id": "a08290a345356a77",
      "type": "ndi_hx",
      "name": "NDI-HX",
      "enable": false,
      "status": "stop",
      "bindAudio": 1,
      "ttl": 1,
      "channel_name": "channel-hdmi",
      "group": "",
      "netmask": "255.255.0.0",
      "connection": "disable_rudp",
      "netprefix": "239.255.0.0",
      "discovery_server": "",
      "auto_enable": false
    }
  ],
  "sdi": [
    {
      "id": "38e19b8c6c15f7c1",
      "type": "ndi_hx",
      "name": "NDI-HX",
      "enable": false,
      "status": "stop",
      "bindAudio": 1,
      "ttl": 1,
      "channel_name": "channel-sdi",
      "group": "",
      "netmask": "255.255.0.0",
      "connection": "disable_rudp",
      "netprefix": "239.255.0.0",
      "discovery_server": "",
      "auto_enable": false
    }
  ],
  "pip": [
    {
      "id": "60c47e02dd3246a2",
      "type": "ndi_hx",
      "name": "NDI-HX",
      "enable": false,
      "status": "stop",
      "bindAudio": 1,
      "ttl": 1,

```

```

    "channel_name": "channel-mix",
    "group": "",
    "netmask": "255.255.0.0",
    "connection": "disable_rudp",
    "netprefix": "239.255.0.0",
    "discovery_server": "",
    "auto_enable": false
  }
]
}

```

* 响应参数说明

参数同流列表，新增参数 auto_enable，类型 Boolean，表示该流是否加入到快捷推流列表

3.4.2 配置快捷推流

* 接口描述

配置快捷推流

* 请求地址

/api/streamer/shortcuts/set

* 请求方法

POST

* 请求参数

```

{
  "panel_enable": true,
  "shortcuts_record": ["hdm", "sdi", "pip"],
  "shortcuts_stream": {
    "hdm": ["d8df2ef3e0b3ca86"],
    "sdi": ["38e19b8c6c15f7c1", "0cc29b8bc74cd729"],
    "pip": ["60c47e02dd3246a2"]
  }
}

```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
shortcuts_record	Array	Y	["hdm", "sdi", "pip"]	加入到快捷录制的

				源
shortcuts_stream	Object	Y	{ "hdmi": ["d8df2ef3e0b3ca86"], "sdi": ["38e19b8c6c15f7c1", "0cc29b8bc74cd729"], "pip": ["60c47e02dd3246a2"] }	加入到快捷推流的流列表
panel_enable	Boolean	N	False	是否开启前面板推流功能。默认为True 开启

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

4 网络服务/Service of network

4.1 网络管理/Network management

4.1.1 获取 eth0 网络信息(PC 端用)

* 接口描述

获取 eth0 网络信息

* 请求地址

/api/networkmanager/network/GetLinkinfo

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "data": [
    {
      "status": "up",
      "address": "192.168.3.37",
      "netmask": "255.255.255.0",
      "gw": "192.168.3.1",
      "mac": "18:C3:F4:20:F1:DB",
      "method": "dhcp",
      "enable": true,
      "dns":
"223.5.5.5;223.6.6.6;101.226.4.6;180.76.76.76;114.114.114.114;119.29.29.29;
192.168.3.1",
      "device": "eth0"
    }
  ]
}
```

* 响应参数说明

字段名	字段类型	字段描述
status	String	当前网络是否被激活 状态由当前网口是否接入网络决定。 "up": 网络激活 "down": 网络未激活
address	String	IP 地址。可能来自 dhcp 或者手动设置。
netmask	String	子网掩码
gw	String	网关
mac	String	MAC 地址
method	String	地址获取方式 dhcp: DHCP 自动获取; static: 手动设置
enable	Boolean	当前网卡是否有效。当前版本该字段值默认为 true
dns	String	DNS, 每个 dns 地址由分号隔开, 返回的是字符串。

device	String	网卡
--------	--------	----

4.1.2 修改当前设备网络信息(全部)

* 接口描述

修改当前设备网络信息

* 请求地址

/api/networkmanager/network/ChangeLinkinfo

* 请求方法

POST

* 请求参数

```
{
  "ifname": "eth0",
  "cfg": {
    "address": "0.0.0.0",
    "netmask": "255.255.255.0",
    "gw": "",
    "mac": "44:07:6A:5F:D4:F7",
    "method": "dhcp",
    "enable": true,
    "dns": "223.5.5.5;224.5.5.5",
  }
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
address	String	Y	192.168.35.35	IP 地址
netmask	String	Y	255.255.255.0	子网掩码
gw	String	Y	192.168.35.254	网关
mac	String	Y	44:07:6A:5F:D4:F7	MAC 地址
method	String	Y	dhcp	地址获取方式, dhcp: DHCP 自动获取; static: 手动设置
enable	Boolean	Y	true	当前网卡是否有效。手动设置
dns	String	N	223.5.5.5;224.5.5.5	dns 地址, 多个由; 区分

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

4.1.3 以太网开关

* 接口描述

用于控制当前的网卡是否激活

* 请求地址

/api/networkmanager/network/SetSwitch

* 请求方法

POST

* 请求参数

```
{  
  "ifname": "eth0",  
  "enable": true  
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
iframe	String	Y	eth0	网卡
enable	Boolean	Y	true	开启/关闭当前网卡

* 响应参数

```
{  
  "result": "ok",  
  "msg": null  
}
```

* 响应参数说明

无

4.1.4 获取网卡信息(小屏用)

* 接口描述

获取网卡信息(小屏用)

* 请求地址

/api/networkmanager/network/GetEthernets

* 请求方法

GET

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
iframe	String	Y	eth0	网卡名称

* 响应参数

```
{
  "result": "ok",
  "data": {
    "status": "up",
    "address": "192.168.3.37",
    "netmask": "255.255.255.0",
    "gw": "192.168.3.1",
    "mac": "18:C3:F4:20:F1:DB",
    "method": "dhcp",
    "enable": true,
    "dns":
"223.5.5.5;223.6.6.6;101.226.4.6;180.76.76.76;114.114.114.114;119.29.29.29;
192.168.3.1",
    "device": "eth0"
  }
}
```

* 响应参数说明

字段名	字段类型	字段描述
status	String	当前网络是否被激活 状态由当前网口是否接入网络决定。 "up": 网络激活 "down": 网络未激活
address	String	IP
netmask	String	子网掩码
gw	String	网关
mac	String	MAC 地址
method	String	地址获取方式 dhcp: DHCP 自动获取; static: 手动设置
enable	Boolean	当前网卡是否开启。当前版本该字段值默认为 true
dns	String	DNS, 每个 dns 地址由分号隔开, 返回的是字符串。
device	String	网卡

4.1.5 设置网卡信息

* 接口描述

设置网卡信息

* 请求地址

/api/networkmanager/network/SetEthernets

* 请求方法

POST

* 请求参数

```
{
  "ifname": "eth0",
  "address": "192.168.38.217",
  "netmask": "255.255.255.0",
  "gw": "192.168.38.254",
  "mac": "56:3D:69:9B:5A:C7",
  "method": "static",
  "dns": "223.5.5.5;224.5.5.5",
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
ifname	String	Y	eth0	网卡名称
address	String	Y	192.168.35.35	IP 地址
netmask	String	Y	255.255.255.0	子网掩码
gw	String	Y	192.168.38.254	网关
mac	String	Y	56:3D:69:9B:5A:C7	MAC 地址
method	String	Y	dhcp	IP 获取方式
dns	String	Y	223.5.5.5;224.5.5.5	DNS 地址

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

4.1.6 重置 MDNS

* 接口描述

立即重启 MDNS

* 请求地址

/api/networkmanager/network/ResetMDNS

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{  
  "result": "ok",  
  "msg": null  
}
```

* 响应参数说明

无

4.2 静态路由管理/Static routing

4.2.1 添加静态路由

* 接口描述

请求此接口添加静态路由地址

* 请求地址

/api/networkmanager/static/routing/add

* 请求方法

POST

* 请求参数

```
{  
  "interface": "eth0",  
  "target": "192.168.38.12",  
  "netmask": "255.255.255.0",  
  "gateway": "192.168.38.254",  
  "mtu": 1500,  
  "metric": 0  
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
interface	String	Y	eth0	网卡
target	String	Y	192.168.42.76	目标 IP 地址
netmask	String	Y	255.255.255.0	子网掩码
gateway	String	Y	192.168.43.254	网关
mtu	Integer	Y	0	路由开销
metric	Integer	Y	1500	最大传输单元

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

4.2.2 获取静态路由列表

* 接口描述

如果设备中已经有保存过之前设置的静态路由列表，则可以请求此接口获取静态路由列表。

* 请求地址

/api/networkmanager/static/routing/get

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "interface": "eth0",
  "target": "192.168.38.12",
  "netmask": "255.255.255.0",
  "gateway": "192.168.38.254",
  "mtu": 1500,
  "metric": 0
}
```

* 响应参数说明

字段名	字段类型	字段描述
interface	String	网卡
target	String	目标
netmask	String	网络掩码
gateway	String	网关
mtu	Integer	最大传输单元
metric	Integer	路由开销
status	Integer	路由启用状态

4.2.3 修改静态路由

* 接口描述

修改指定的静态路由

* 请求地址

/api/networkmanager/static/routing/modify

* 请求方法

POST

* 请求参数

```
{
  "id":1,
  "interface":"eth0",
  "target":"192.168.38.12",
  "netmask":"255.255.255.0",
  "gateway":"192.168.38.254",
  "mtu":1500,
  "metric":0
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
id	String	Y	1	路由列表 ID
interface	String	Y	eth0	网卡
target	String	Y	192.168.35.22	目标 IP
netmask	String	Y	255.255.255.0	子网掩码
gateway	String	Y	192.168.38.254	网关
mtu	Integer	Y	0	最大传输单元
metric	Integer	Y	1500	路由开销

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

4.2.4 删除静态路由

* 接口描述

请求此接口可删除单个静态路由，或者批量删除部分静态路由

* 请求地址

/api/networkmanager/static/routing/del

* 请求方法

POST

* 请求参数

```
// 删除单个
{
  "ids": 1
}
// 删除多个
{
  "ids": [1, 2]
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
ids	Array	Y	[1]	路由 id 或 id 数组

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

4.2.5 静态路由开关

* 接口描述

通过请求该接口，可以开启/关闭指定路由

* 请求地址

/api/networkmanager/static/routing/set

* 请求方法

POST

* 请求参数

```
{
  "id": 1,
  "status": 1
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
------	------	------	-------	----

id	Integer	Y	1	路由 ID
status	Integer	Y	1	开启/关闭指定路由

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

5 Kilolink 客户端/Service of kilolink client

5.1 kilolink/kilolink

5.1.1 获取当前设备网络接口

* 接口描述

获取当前设备网络接口,等小屏修改完后弃用

* 请求地址

/api/kilolink/networks

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "list": ["eth0"]
}
```

* 响应参数说明

字段名	字段类型	字段描述
list	Array	网络接口列表

5.1.2 获取 kilolink client 设置

* 接口描述

获取 kilolink client 设置

* 请求地址

/api/kilolink/get

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "cfg": {
    "ip": "",
    "port": 50000,
    "ifname": [],
    "key": "",
    "crypto": false,
    "enable": false,
  },
  "status": 1
}
```

* 响应参数说明

字段名	字段类型	字段描述
ip	String	服务器地址
port	Integer	服务端口
iframe	Array	网络接口
key	String	授权码
crypto	Boolean	是否加密
enable	Boolean	是否开启
status	Integer	连接状态

5.1.3 设置 kilolink client 配置

* 接口描述

设置 kilolink client 配置

* 请求地址

/api/kilolink/set

* 请求方法

POST

* 请求参数

```
{
  "cfg":{
    "enable":false,
    "key":"ASdq41a",
    "ip":"192.168.1.40",
    "port":50000,
    "crypto":false,
    "ifname":["eth0"]
  }
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
ip	String	Y	192.168.1.40	服务器地址
port	Integer	Y	5000	服务端口
iframe	Array	Y	["eth0"]	网络接口
key	String	Y		授权码
crypto	Boolean	Y	false	是否加密
enable	Boolean	Y	false	是否开启

* 响应参数

```
{
  "result": "ok"
}
```

* 响应参数说明

无

6 编码服务/Service of codec

6.1 视频输入/Video in

6.1.1 获取 Edid 信息

* 接口描述

获取 EDID 信息

* 请求地址

/api/codec/vin/edid

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "edid_priority": "2",
  "edid":
  "AP////////wAmhQMAQhYDAAshAQOAFp46mehpVVNoicOUFQhCAABAQEBAQEBAQEBAQEBAQEBBH
  QAMPJwWoCwWIoAIMIxAAAeAAAAEAAAAAAAAAAAAAAAAAAAAAAAAAA/ABFMy1FTkNPREVSCiAgAAAAE
  AAAAAAAAAAAAAAAAAAAAdICAznxVF9eXZAFiiEgBRQEEz49PAMHFhIBIwsHB4MQAABtAwWAEAAI
  ACAAYAECA2fYXcQBeIAD4Q8AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAg==",
  "reset": false
}
```

* 响应参数说明

字段名	字段类型	字段描述
edid	String	直接返回纯文本的 EDID 信息，无需 JSON 解析 EDID 信息，base64 格式，显示时需要转换成 16 进制值。
edid_priority	String	优先级。 1:1080p, 2:4k, "":无优先级
reset	Boolean	重置按钮是否可用

6.1.2 上传 Edid 信息

* 接口描述

上传 EDIT 信息

* 请求地址

/api/codec/vin/edid/upload

* 请求方法

POST

* 请求参数

无

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
file	Binary	Y		二进制文件

* 响应参数

```
{
  "result": "ok",
```

```
"msg": null
}
```

* 响应参数说明

无

6.1.3 设置 edid 优先级

* 接口描述

重置 EDIT 信息

* 请求地址

/api/codec/vin/edid/priority

* 请求方法

POST

* 请求参数

```
{
  "edid_priority": "1"
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
edid_priority	String	Y	1	优先级。 1:1080p, 2:4k, "":无优先级

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

6.1.4 重置 Edid 信息

* 接口描述

重置 EDIT 信息

* 请求地址

/api/codec/vin/edid/reset

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{  
  "result": "ok",  
  "msg": null  
}
```

* 响应参数说明

无

6.1.5 获取开启的编码

* 接口描述

获取开启的编码接口

* 请求地址

/api/codec/vin/active_interface

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{  
  "hdmi": true,  
  "sdi": false,  
  "pip": false,  
  "uvc": false  
}
```

* 响应参数说明

字段名	字段类型	字段描述
hdmi	Boolean	hdmi 编码是否开启
sdi	Boolean	sdi 编码是否开启
pip	Boolean	画中画编码是否开启
uvc	Boolean	uvc 编码是否开启

6.1.6 获取视频源信息

* 接口描述

获取接入源的信息。

* 请求地址

/api/codec/{interface}/vin/detail

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "signal": "locked",
  "format": "1920x1080p 60Hz",
  "width": 1920,
  "height": 1080,
  "interlaced": false,
  "fps": 60,
  "type": "HDMI",
  "interface": "hdmi",
  "max_width": 3840,
  "max_height": 2160,
  "max_fps": 60
}
```

* 响应参数说明

字段名	字段类型	字段描述
signal	String	当前信号状态，locked 表示信号已接入，none 表示未接入信号
format	String	视频源接入格式
width	Boolean	视频源宽度
height	Integer	视频源高度
interlaced	Integer	视频源图像是否交错
fps	Boolean	帧率
type	String	HDMI 或 SDI，视频源接入类型
interface	String	hdmi 或 sdi，视频源接入接口
max_width	Integer	当前源最大宽度
max_height	Integer	当前源最大高度

max_fps	Integer	当前源最大帧率
vertical	Boolean	源是否为竖屏分辨率
input_fps	Integer	输入源帧率。同 fps
input_encode	String	输入编码格式，h264,h265,mjpeg
input_resolution	String	输入分辨率

6.1.7 获取视频格式列表

* 接口描述

获取当前视频格式列表，仅 uvc

* 请求地址

/api/codec/{interface}/vin/list_format

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
[
  {
    "label": "MJPEG",
    "value": "mjpeg",
    "children": [
      {
        "label": "1920X1080",
        "value": "1920x1080",
        "children": [
          {
            "label": 60,
            "value": 60
          },
          {
            "label": 30,
            "value": 30
          }
        ]
      }
    ]
  },
  {
    "label": "1280X720",
    "value": "1280x720",
    "children": [
      {
        "label": 60,
        "value": 60
      }
    ]
  }
]
```



```

"input_resolution": "1920x1080",
"input_fps": 24
}
pip:
{
"width": 1920,
"height": 1080,
"input_fps": 24
}

```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
input_encode	String	N	mjpeg	输入编码格式，可选，取值范围为 h264,h265,mjpeg
input_resolution	String	N	1920x1080	输入分辨率，可选，如 1920x1080。和 width、height 二选一填入即可
input_fps	Integer	N	24	输入帧率，可选，如 30
width	Integer	N	1920	画面宽度，可选
height	Integer	N	1080	画面高度，可选

* 响应参数

```

{
"result": "ok",
"msg": null
}

```

* 响应参数说明

无

6.2 音频/Audio

6.2.1 获取输入通道的音频来源信息

* 接口描述

用于获取输入通道的音频来源信息

* 请求地址

/api/codec/{interface}/audio/input

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "source": "hdmi",
  "sampling": 44100,
  "channels": 2,
  "aichannel": 0,
  "gain": -20
}
```

* 响应参数说明

字段名	字段类型	字段描述
source	String	音频源, hdmi/sdi/line-in/uvc
sampling	Integer	音频输入源的音频采样率
channels	Integer	音频输入源的音频通道数
aichannel	Integer	使用音频通道, 0 表示 1/2, 2 表示 3/4, 3 表示 5/6 以此类推
gain	Integer	音频增益的值, 范围-40 ~ 20

6.2.2 选择音频输入的来源

* 接口描述

选择音频输入的来源

* 请求地址

/api/codec/{interface}/audio/input/source

* 请求方法

POST

* 请求参数

```
{
  "source": "string",
  "gain": 0,
  "mute": false
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
source	String	Y	sdi	音频输入来源, hdmi/sdi/linein/uvc
gain	Integer	N	0	音频增益的值, 范围为-40 ~ 20
mute	Boolean	N	false	是否静音。true 表示开启静音

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

6.2.3 获取音频编码列表

* 接口描述

获取音频编码列表

* 请求地址

/api/codec/{interface}/audio/encode/list

* 请求方法

POST

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
[
  {
    "name": "channel-1",
    "volume": 0,
    "resample": "fastest",
    "channels": 2,
    "sampleSize": 1024,
    "sampling": 48000,
    "codec": "AAC",
    "bitrate": 64000,
    "sampleWidth": 16,
    "g711Law": "PCMU",
    "id": 1
  },
  {
    "name": "channel-2",
    "volume": 0,
    "resample": "fastest",
    "channels": 1,
    "sampleSize": 1024,
    "sampling": 8000,
    "codec": "G711",
    "bitrate": 64000,
    "sampleWidth": 16,
    "g711Law": "PCMU",
    "id": 2
  }
]
```

* 响应参数说明

字段名	字段类型	字段描述
id	Integer	音频编码 ID, 唯一 ID
name	String	音频编码名称
resample	String	G.711 的重采样策略, common (高品质)、fastest (一般)
channels	Integer	音频通道数, G.711 的音频通道数固定为 1
sampling	Integer	音频采样率
codec	String	音频编码格式, AAC、G.711、MP2、MP3
bitrate	Integer	编码码率, 单位 kbps
g711law	String	G.711 音频格式, PCMU (u-Law)、PCMA (a-Law)

6.2.4 添加音频编码

* 接口描述

添加音频编码

* 请求地址

/api/codec/{interface}/audio/encode

* 请求方法

POST

* 请求参数

```
{
  "name": "string",
  "codec": "string",
  "sampling": 0,
  "bitrate": 524288,
  "channels": 0,
  "resample": "string",
  "g711law": "string",
  "track_num": 1
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
name	String	Y	666	音频编码名
codec	String	Y	AAC	音频编码格式, AAC/G711/MP2/MP3
sampling	String	N	8000	音频采样率

bitrate	Integer	N	64000	音频编码码率
channels	Integer	N	2	音频通道数
resample	String	N	fastest	G.711 重采样策略
g711Law	String	N	PCMU	G.711 格式, PCMU (u-Law)、PCMA (a-Law)
track_num	Integer	N	1	音轨数量

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

6.2.5 修改音频编码

* 接口描述

修改音频编码

* 请求地址

/api/codec/{interface}/audio/encode/{audio_id}

* 请求方法

PUT

* 请求参数

```
{
  "id": 1,
  "name": "123",
  "codec": "G711",
  "resample": "fastest",
  "sampling": 8000,
  "channels": 1,
  "bitrate": 64000,
  "g711Law": "PCMA",
  "track_num": 1
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
id	Integer	Y	1	音频编码唯一 ID, 用于指定修改哪个音频编码设置
name	String	Y	666	音频编码名称

codec	String	Y	G711	音频编码格式
resample	String	Y	fastest	G.711 重采样策略
sampling	Integer	Y	8000	音频采样率
channels	Integer	Y	1	音频通道数
bitrate	Integer	Y	64000	音频编码码率
g711Law	String	Y	PCMU	G.711 音频格式, PCMU (u-Law)、PCMA (a-Law)
track_num	Integer	N	1	音轨数量

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

6.2.6 删除音频编码

* 接口描述

删除音频编码配置

* 请求地址

/api/codec/{interface}/audio/encode/{audio_id}

* 请求方法

POST

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

6.2.7 获取音频源列表

* 接口描述

获取当前接口的音频源的列表

* 请求地址

/api/codec/{interface}/audio/input/list_source

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
[
  {
    "label": "HDMI",
    "value": "hdmi"
  },
  {
    "label": "SDI",
    "value": "sdi"
  },
  {
    "label": "Line In",
    "value": "linein"
  },
  {
    "label": "USB Camera",
    "value": "uvc"
  }
]
```

* 响应参数说明

字段名	字段类型	字段描述
label	String	音频源名称
value	String	音频源接口名称

6.3 视频编码/Video encode

6.3.1 获取当前编码参数配置

* 接口描述

获取当前编码参数配置

* 请求地址

/api/codec/{interface}/venc/settings

* 请求方法

GET

* 请求参数

venc_type=main

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
Interface	String	Y	hdmi	hdmi/sdi/pip/uvc, 表示开启的对应编码
venc_type	String	N	main	main/sub/both, 表示获取主码流或子码流的编码参数。不填默认为 main

* 响应参数

```
{
  "codec": "H264",
  "enable": true,
  "grey": false,
  "profile": 0,
  "picHeight": 1,
  "picWidth": 1,
  "mode": "CBR",
  "gop": 60,
  "fps": 60,
  "real_fps": 60,
  "real_fps_float": 59.94,
  "bitrate": 20000,
  "maxBitrate": 6000,
  "minQp": 24,
  "maxQp": 51,
  "minIqp": 24,
  "minIp": 1,
  "maxIp": 20,
  "minQfactor": 80,
  "maxQfactor": 85,
  "deinterlaceMode": "weave",
  "srcHeight": 1080,
  "srcWidth": 1920,
  "realWidth": 1920,
  "realHeight": 1080,
  "timecode_type": 0
}
```

* 响应参数说明

字段名	字段类型	字段描述
codec	String	视频编码格式, H264 或 H265
enable	Boolean	是否开启

grey	Boolean	画面色彩是否展示灰度图像
profile	Integer	视频档次, 0:Baseline 1:Main Profile 2: High Profile
picHeight	Integer	图像高度, 注: picHeight 与 picWidth 均为 1 则表示与输入源的宽高一致
picWidth	Integer	图像宽度, 注: picHeight 与 picWidth 均为 1 则表示与输入源的宽高一致
mode	String	码率模式, CBR 恒定码率模式, VBR 动态码率模式
gop	Integer	gop 大小
fps	Integer/String	帧率, 当自定义帧率时, 设置一个 INT 整型数据, 即指定码率。如果为 STRING 请传入例如 auto (自动全帧率)、half (帧率>=50 时自动编码帧率减半)
real_fps	Integer	用户设置的真实的帧率, 当帧率设置 auto、half 模式时, 对应的真实帧率
real_pfs_float	Float	用户设置的真实的带浮点的帧率, 当帧率设置 auto、half 模式时, 对应的真实帧率。为了能正常显示小数帧率, 例如 59.94、29.97、23.98, 使用户清楚的知道正在编码的真实小数帧率
bitrate	Integer	CBR 恒定编码码率, 单位为 kb/s
maxBitrate	Integer	VBR 动态码率, 最大码率大小, 单位为 kb/s
minQp	Integer	动态码率模式最小 Qp 值
maxQp	Integer	动态码率模式最大 Qp 值
minIQp	Integer	动态码率模式最小 IQp 值
minIqp	Integer	固定码率模式 Iqp 帧比例范围的最小值
maxIqp	Integer	固定码率模式 Iqp 帧比例范围的最大值
minQfactor	Integer	mjpeg 编码格式下图像质量范围的最小值
maxQfactor	Integer	mjpeg 编码格式下图像质量范围的最大值
deinterlaceMode	String	weave/line_doubling, 去隔行扫描模式
srcHeight	Integer	接入源视频高度
srcWidth	Integer	接入源视频宽度
realWidth	Integer	真实编码视频宽度
realHeight	Integer	真实编码视频高度
timecode_type	Integer	时间码类型 0 表示关闭时间码, 1 表示系统时间, 2 表示输入源时间 (暂不支持)

6.3.2 修改编码参数

* 接口描述

修改当前视频编码参数

* 请求地址

/api/codec/{interface}/venc/settings

* 请求方法

POST

* 请求参数

```
{
  "venc_type": "main",
  "codec": "H264",
  "grey": false,
  "profile": 0,
  "picHeight": 1,
  "picWidth": 1,
  "mode": "CBR",
  "gop": 60,
  "fps": 60,
  "bitrate": 3000,
  "maxBitrate": 6000,
  "minQp": 24,
  "maxQp": 51,
  "minIQp": 24,
  "minQfactor": 80,
  "maxQfactor": 85,
  "deinterlaceMode": "line_doubling",
  "timecode_type": 0
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
venc_type	String	N	main	main/sub, 表示设置主码流或子码流的编码参数。不填默认为 main
codec	String	Y	H264	视频编码格式, H264/H265/MJPEG
grey	Boolean	Y	false	是否使图像做灰度处理
fps	Integer	Y	60	帧率大小
picHeight	Integer	Y	1	图像高度, 注: picHeight 与 picWidth 均为 1 则表示与输入源的宽高一致
picWidth	Integer	Y	1	图像宽度, 注: picHeight 与 picWidth 均为 1 则表示与输入源的宽高一致

mode	String	N	CBR	码率模式
profile	Integer	N	0	视频档 次 0:Baseline 1:Main Profile 2: High Profile 注意视频编码格式非 H264 时，该值只能为 0
gop	Integer	N	60	gop 大小
bitrate	Integer	N	3000	CBR 恒定编码码率，单位为 kb/s
maxBitrate	Integer	N	6000	VBR 动态码率，最大码率大小，单位为 kb/s
minQp	Integer	N	24	动态码率模式最小 Qp 值
maxQp	Integer	N	51	动态码率模式最大 Qp 值
minIQp	Integer	N	24	动态码率模式最小 IQp 值
minIp	Integer	N	1	CBR 模式最小 Ip 帧比例
maxIp	Integer	N	20	CBR 模式最大 Ip 帧比例
minQfactor	Integer	N	80	mjpeg 编码格式下图像质量范围的最小值
maxQfactor	Integer	N	85	mjpeg 编码格式下图像质量范围的最大值
deinterlaceMode	String	N	line_doubling	去隔行扫描模式，line_doubling/weave
timecode_type	Integer	N	0	时间码类型 0 表示关闭时间码，1 表示系统时间，2 表示输入源时间（暂不支持）

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

6.3.3 设置编码码率

* 接口描述

设置编码码率参数

* 请求地址

/api/codec/{interface}/venc/bitrate

* 请求方法

POST

* 请求参数

```
{  
  "venc_type": "main",  
  "bitrate": 102400  
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
venc_type	String	N	main	main/sub, 表示设置主码流或子码流的码率。不填默认为 main
bitrate	Integer	Y	102400	CBR 恒定编码码率, 单位为 kb/s

* 响应参数

```
{  
  "result": "ok",  
  "msg": null  
}
```

* 响应参数说明

无

6.3.4 编码开关

* 接口描述

该接口用于关闭彻底关闭接入源的对应视频、音频编码。

* 请求地址

/api/codec/{interface}/venc/enable

* 请求方法

POST

* 请求参数

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

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
enable	Boolean	Y	true	编码开关

* 响应参数

```
{  
  "result": "ok",  
  "msg": ""  
}
```

* 响应参数说明

无

6.3.5 切换画中画主子

* 接口描述

该接口用于切换画中画以什么画面为主画面

* 请求地址

/api/codec/pip/venc/pip_back_update

* 请求方法

POST

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{  
  "result": "ok",  
  "msg": ""  
}
```

* 响应参数说明

无

6.3.6 获取画中画布局

* 接口描述

该接口用于获取画中画的布局参数

* 请求地址

/api/codec/pip/venc/pip_layout_setting

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
[
  {
    "height": 100,
    "width": 100,
    "name": "hdmi",
    "x": 0,
    "y": 0,
    "zIndex": 1,
    "enable": true
  },
  {
    "height": 33.33,
    "width": 33.33,
    "name": "sdi",
    "x": 65,
    "y": 64,
    "zIndex": 2,
    "enable": true
  }
]
```

* 响应参数说明

字段名	字段类型	字段描述
name	String	所在视频源名称，hdmi/sdi
width	Float	画面的图像宽度
height	Float	画面的图像高度
x	Integer	画面的 x 坐标，按百分比计算
y	Integer	画面的 y 坐标，按百分比计算
zIndex	Integer	画面层级，数字越大层级越高
enable	Boolean	当前视频源画面是否显示

6.3.7 设置画中画布局

* 接口描述

该接口用于设置画中画的布局

* 请求地址

/api/codec/pip/venc/pip_layout_setting

* 请求方法

POST

* 请求参数

```
[
  {
    "height": 100,
    "width": 100,
    "name": "hdmi",
    "x": 0,
    "y": 0,
    "zIndex": 1,
    "enable": true
  },
  {
    "height": 33.33,
    "width": 33.33,
    "name": "sdi",
    "x": 65,
    "y": 64,
    "zIndex": 2,
    "enable": true
  }
]
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
name	String	Y	hdmi	所在视频源名称, hdmi/sdi
x	Float	Y	33.33	画面的图像宽度
y	Float	Y	33.33	画面的图像高度
width	Integer	Y	50	画面的 x 坐标, 按百分比计算
height	Integer	Y	50	画面的 y 坐标, 按百分比计算
zIndex	Integer	Y	1	画面层级, 数字越大层级越高
enable	Boolean	Y	true	当前视频源画面是否显示

* 响应参数

```
{
  "result": "ok",
  "msg": ""
}
```

* 响应参数说明

无

6.3.8 获取画中画视频源的帧率&分辨率

* 接口描述

该接口用于获取画中画的布局参数

* 请求地址

/api/codec/pip/venc/pip_fps_setting

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "fps": 50,
  "width": 1920,
  "height": 1080,
  "vertical": false
}
```

* 响应参数说明

字段名	字段类型	字段描述
fps	Integer	设定的视频源帧率。可选 24/25/30/50/60
width	Integer	设定的视频源宽度
height	Integer	设定的视频源高度
vertical	Boolean	当前是否为竖屏分辨率。若 height>width 则为 true

6.3.9 设置画中画视频源的帧率&分辨率

* 接口描述

该接口用于设置画中画的布局

* 请求地址

/api/codec/pip/venc/pip_fps_setting

* 请求方法

POST

* 请求参数

```
{
  "fps": 50,
  "width": 1920,
  "height": 1080
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
fps	Integer	N	50	设定的视频源帧率。可选 24/25/30/50/60

width	Integer	N	1920	设定的视频源宽度
height	Integer	N	1080	设定的视频源高度

* 响应参数

```
{
  "result": "ok",
  "msg": ""
}
```

* 响应参数说明

无

6.3.10 获取子码流配置是否符合 NDI 标准

* 接口描述

该接口用于获取当前子码流设置是否符合 NDI|HX 协议标准。NDI|HX 子码流分辨率不能超过 640*360，并且不能使用 MJPEG 编码格式。

* 请求地址

/api/codec/{interface}/venc/complie_NDIstandard

* 请求方法

GET

* 请求参数

无

* 响应参数

```
{
  "is_standard": True,
  "ndi_hx_is_streaming": False
}
```

* 响应参数说明

字段名	字段类型	字段描述
is_standard	Boolean	当前子码流设置是否符合 NDI\
is_ndi_hx_is_streaming	Boolean	当前接口的 ndi 流是否开启，为 true 则开启

6.3.11 恢复子码流为符合 NDI 标准的配置

* 接口描述

该接口用于恢复子码流为符合 NDI 标准的配置

* 请求地址

/api/codec/{interface}/venc/complie_NDIstandard

* 请求方法

POST

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{  
  "result": "ok",  
  "msg": ""  
}
```

* 响应参数说明

无

6.3.12 获取当前编码快照

* 接口描述

子码流设置为 MJPEG 时，可以调用该接口获取当前的编码快照。如果编码未开启或者子码流未设置为 MJPEG 编码格式，则返回默认的黑底白字提示图片。

* 请求地址

/api/codec/{interface}/venc/snap

* 请求方法

GET

* 请求参数

无

* 响应参数

image/jpeg

* 响应参数说明

无

6.4 视频处理/VPSS

6.4.1 获取图片列表

* 接口描述

用于获取图文叠加中的图片列表

* 请求地址

/api/codec/vpss/image

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
[
  {
    "size_h": 276,
    "size_w": 342,
    "url": "/img/3f33d7027c5111eea32f180dab8597db.png",
    "name": "123",
    "path": "/data/storage/img/3f33d7027c5111eea32f180dab8597db.png",
    "id": "3f33d7027c5111eea32f180dab8597db"
  }
]
```

* 响应参数说明

字段名	字段类型	字段描述
id	String	图片唯一 ID
size_h	Integer	图片尺寸高度
size_w	Integer	图片尺寸宽度
url	String	拼接 host 即可获取静态图片资源
name	String	图片名称
path	String	Unix 文件系统路径

6.4.2 上传图片

* 接口描述

上传图片

* 请求地址

/api/codec/vpss/image

* 请求方法

POST

* 请求参数

```
{
  "file": (binary)
  "name": 456
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
file	Binary	Y		图片文件
name	String	Y	123	图片名称

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

6.4.3 删除图片

* 接口描述

删除图片

* 请求地址

/api/codec/vpss/image/{image_id}

* 请求方法

DELETE

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

6.4.4 获取 Vpss 对象视频输入的裁切参数

* 接口描述

获取 Vpss 对象视频输入的裁切参数

* 请求地址

/api/codec/{interface}/vpss/input/crop

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{  
  "crop_en": true,  
  "crop_x": 25,  
  "crop_y": 25,  
  "crop_w": 50,  
  "crop_h": 50  
}
```

* 响应参数说明

字段名	字段类型	字段描述
crop_en	Boolean	裁切是否开启。为 true 时下面的参数才会生效
crop_x	Integer	裁切起始 x 坐标，百分比计算
crop_y	Integer	裁切起始 y 坐标，百分比计算
crop_w	Integer	裁切画面宽度，百分比计算
crop_h	Integer	裁切画面高度，百分比计算

6.4.5 设置 Vpss 对象视频输入的裁切参数

* 接口描述

修改 VPSS 对象视频输入的裁切参数。目前只有在画中画模式开启的时候调用生效，可分别裁剪 hdmi/sdi 视频源。

* 请求地址

/api/codec/{interface}/vpss/input/crop

* 请求方法

POST

* 请求参数

```
{  
  "crop_en": true,  
  "crop_x": 25,  
  "crop_y": 25,  
  "crop_w": 50,  
  "crop_h": 50  
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
crop_en	Boolean	Y	true	裁切是否开启。为 true 时下面的参数才会生效
crop_x	Integer	Y	25	裁切起始 x 坐标，百分比计算
crop_y	Integer	Y	25	裁切起始 y 坐标，百分比计算
crop_w	Integer	Y	25	裁切画面宽度，百分比计算
crop_h	Integer	Y	25	裁切画面高度，百分比计算

* 响应参数

```
{
  "result": "ok",
  "msg": ""
}
```

* 响应参数说明

无

6.4.6 获取 Vpss 对象视频输入的旋转，镜像，翻转参数

* 接口描述

获取 VPSS 对象视频输入的旋转、镜像、翻转参数。

* 请求地址

/api/codec/{interface}/vpss/input/rmf

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "rotate": 0,
  "mirror_en": false,
  "flip_en": false
}
```

* 响应参数说明

字段名	字段类型	字段描述
rotate	Integer	旋转角度，0/90/180/270

mirror_en	Boolean	镜像开关
flip_en	Boolean	翻转开关

6.4.7 设置 Vpss 对象视频输入的旋转，镜像，翻转参数

* 接口描述

设置 VPSS 对象视频输入的旋转、镜像、翻转参数

* 请求地址

/api/codec/{interface}/vpss/input/rmf

* 请求方法

POST

* 请求参数

```
{
  "rotate": 0,
  "mirror_en": false,
  "flip_en": false
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
rotate	String	Y	0	旋转角度，0/90/180/270
mirror_en	Boolean	Y	false	镜像开关
flip_en	Boolean	Y	false	翻转开关

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

6.4.8 获取 Osd 叠加列表

* 接口描述

获取 OSD 叠加列表

* 请求地址

/api/codec/{interface}/vpss/osd

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "enable": true,
  "items": [
    {
      "enable": true,
      "type": "image",
      "position": "LeftTop",
      "x": 0,
      "y": 0,
      "font_size": 1,
      "font_color": 1,
      "style": "normal",
      "outline": 0,
      "outline_color": 0,
      "text": "",
      "file_id": "66bd63907c5311eea32f180dab8597db",
      "row_spacing": 1,
      "id": "830a6f927c6d11eea32f180dab8597db",
      "name": "456",
      "pic_url": "/img/66bd63907c5311eea32f180dab8597db.png"
    },
    {
      "enable": true,
      "type": "text",
      "position": "LeftTop",
      "x": 0,
      "y": 0,
      "font_size": 1,
      "font_color": 1,
      "style": "normal",
      "outline": 0,
      "outline_color": 0,
      "text": "12313",
      "file_id": "",
      "row_spacing": 1,
      "id": "897d2f727c6d11eea32f180dab8597db"
    }
  ]
}
```

* 响应参数说明

字段名	字段类型	字段描述
id	String	叠加项的唯一 ID
enable	Boolean	是否启用该叠加项

type	String	叠加项类型, image 表示图片叠加, time 表示时间和日期叠加, date 表示日期叠加, clock 表示时间叠加, text 表示文本叠加
position	String	叠加项位置, LeftTop 表示左上, CenterTop 表示上边居中, RightTop 表示右上角, LeftBottom 表示左下角, CenterBottom 表示下边居中, RightBottom 表示右下角, Center 表示正中心, Manual 表示自定义
x	Integer	叠加项的 x 坐标偏移值, position 为 Manual 时生效
y	Integer	叠加项的 y 坐标偏移值, position 为 Manual 时生效
font_size	Integer	叠加项的字体大小, 0 表示自动-大字体, 1 表示自动-中等, 2 表示自动-较小, 其他值表示具体像素大小
font_color	Integer	叠加项的字体颜色, 0 表示黑色, 1 表示白色, 2 表示红色, 3 表示蓝色, 4 表示绿色, 5 表示黄色, 6 表示紫色, 7 表示青色
style	String	叠加项的字体样式, normal 表示正常, bold 表示加粗
outline	Integer	叠加项外边框, 0 是无边框, 1 是细边框, 2 是粗边框
outline_color	Integer	叠加项外边框颜色, 0 表示黑色, 1 表示白色, 2 表示红色, 3 表示蓝色, 4 表示绿色, 5 表示黄色, 6 表示紫色, 7 表示青色
text	String	叠加项文本内容, 文本
file_id	String	叠加项关联的文件 id
name	String	叠加项的图片名称
pic_url	String	叠加项的静态图片资源地址

6.4.9 Osd 总开关

* 接口描述

开启或关闭图文叠加选项

* 请求地址

/api/codec/{interface}/vpss/osd

* 请求方法

PUT

* 请求参数

```
{
  "enable": true
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
enable	Boolean	Y	true	图文叠加开关

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

6.4.10 添加 Osd 信息

* 接口描述

添加 OSD 叠加项

* 请求地址

/api/codec/{interface}/vpss/osd

* 请求方法

POST

* 请求参数

```
// 文本叠加:
{
  "type": "text",
  "file_id": "",
  "text": "2222222222",
  "x": 0,
  "y": 0,
  "position": "LeftTop",
  "font_size": 1,
  "style": "normal",
  "outline": 0,
  "font_color": 1,
  "outline_color": 0,
  "enable": true
}

// 日期和时间叠加:
{
  "id": "",
  "type": "time",
  "file_id": "",
  "text": "",
  "x": 0,
  "y": 0,
  "position": "LeftTop",
  "font_size": 1,
  "style": "normal",
```

```
"outline": 0,  
"font_color": 1,  
"outline_color": 0,  
"enable": true  
}  
  
// 日期叠加:  
  
{  
  "id": "",  
  "type": "date",  
  "file_id": "",  
  "text": "",  
  "x": 0,  
  "y": 0,  
  "position": "LeftTop",  
  "font_size": 1,  
  "style": "normal",  
  "outline": 0,  
  "font_color": 1,  
  "outline_color": 0,  
  "enable": true  
}  
  
// 时间叠加:  
  
{  
  "id": "",  
  "type": "clock",  
  "file_id": "",  
  "text": "",  
  "x": 0,  
  "y": 0,  
  "position": "LeftTop",  
  "font_size": 1,  
  "style": "normal",  
  "outline": 0,  
  "font_color": 1,  
  "outline_color": 0,  
  "enable": true  
}  
  
// 图片叠加:  
  
{  
  "id": "",  
  "type": "image",  
  "file_id": "66bd63907c5311eea32f180dab8597db",  
  "text": "",  
  "x": 0,  
  "y": 0,  
  "position": "LeftTop",  
  "font_size": 1,  
  "style": "normal",  
  "outline": 0,  
  "font_color": 1,  
  "outline_color": 0,  
  "enable": true  
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值 样例	描述
enable	Boolean	Y	true	是否开启该叠加项
type	String	Y	text	叠加类型, image 表示图片叠加, time 表示时间和日期叠加, date 表示日期叠加, clock 表示时间叠加, text 表示文本叠加
file_id	String	Y		仅图片叠加时需要传值, 传图片 ID
text	String	Y	123	仅文本叠加时需要传值, 文本内容
x	Integer	Y	0	叠加时自定义的位置, x 坐标
y	Integer	Y	0	叠加时自定义的位置, y 坐标
position	String	Y	LeftTop	叠加时的位置
font_size	Integer	Y	0	叠加项的字体大小, 0 表示自动-大字体, 1 表示自动-中等, 2 表示自动-较小, 其他值表示具体像素大小
style	String	Y	normal	叠加时的文字样式, normal 表示正常, bold 表示加粗
outline	Integer	Y	0	叠加的外边框
font_color	Integer	Y	1	叠加项的字体颜色, 0 表示黑色, 1 表示白色, 2 表示红色, 3 表示蓝色, 4 表示绿色, 5 表示黄色, 6 表示紫色, 7 表示青色
outline_color	Integer	Y	1	外边框的颜色, 0 表示黑色, 1 表示白色, 2 表示红色, 3 表示蓝色, 4 表示绿色, 5 表示黄色, 6 表示紫色, 7 表示青色

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

6.4.11 获取 Osd 叠加素材信息

* 接口描述

获取单个 OSD 叠加项信息

* 请求地址

/api/codec/{interface}/vpss/osd/{osd_id}

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "enable": true,
  "type": "text",
  "position": "Manual",
  "x": 200,
  "y": 200,
  "font_size": 1,
  "font_color": 3,
  "style": "bold",
  "outline": 2,
  "outline_color": 5,
  "text": "12313",
  "file_id": "3f33d7027c5111eea32f180dab8597db",
  "row_spacing": 1,
  "id": "897d2f727c6d11eea32f180dab8597db"
}
```

* 响应参数说明

字段名	字段类型	字段描述
enable	Boolean	是否启用该叠加项
type	String	叠加项类型, image 表示图片叠加, time 表示时间和日期叠加, date 表示日期叠加, clock 表示时间叠加, text 表示文本叠加
position	String	叠加项位置, LeftTop 表示左上, CenterTop 表示上边居中, RightTop 表示右上角, LeftBottom 表示左下角, CenterBottom 表示下边居中, RightBottom 表示右下角, Center 表示正中心, Manual 表示自定义
x	Integer	叠加项的 x 坐标偏移值, position 为 Manual 时生效
y	Integer	叠加项的 y 坐标偏移值, position 为 Manual 时生效
font_size	Integer	叠加项的字体大小, 0 表示自动-大字体, 1 表示自动-中等, 2 表示自动-较小, 其他值表示具体像素大小
font_color	Integer	叠加项的字体颜色, 0 表示黑色, 1 表示白色, 2 表示红色, 3 表示蓝色, 4 表示绿色, 5 表示黄色, 6 表示紫色, 7 表示青

		色
style	String	叠加项的字体样式，normal 表示正常，bold 表示加粗
outline	Integer	叠加项外边框，0 是无边框，1 是细边框，2 是粗边框
outline_color	Integer	叠加项外边框颜色，0 表示黑色，1 表示白色，2 表示红色，3 表示蓝色，4 表示绿色，5 表示黄色，6 表示紫色，7 表示青色
text	String	叠加项文本内容，文本
file_id	String	叠加项关联的文件 id
name	String	叠加项的图片名称
pic_url	String	叠加项的静态图片资源地址
id	String	叠加项的唯一 ID

6.4.12 修改 Osd 素材信息

* 接口描述

修改 OSD 素材信息

* 请求地址

/api/codec/{interface}/vpss/osd/{osd_id}

* 请求方法

PUT

* 请求参数

```
{
  "id": "897d2f727c6d11eea32f180dab8597db",
  "type": "text",
  "file_id": "3f33d7027c5111eea32f180dab8597db",
  "text": "12313666",
  "x": 200,
  "y": 200,
  "position": "Manual",
  "font_size": 1,
  "style": "bold",
  "outline": 2,
  "font_color": 3,
  "outline_color": 5,
  "enable": true
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
id	String	Y	897d2f727c6d11eea32f180dab	指定修改某个叠加

			8597db	项
type	String	Y	text	叠加类型，image 表示图片叠加，time 表示时间和日期叠加，date 表示日期叠加，clock 表示时间叠加，text 表示文本叠加
file_id	String	Y		仅图片叠加时需要传值，传图片 ID
text	String	Y	123	仅文本叠加时需要传值，文本内容
x	Integer	Y	0	叠加时自定义的位置，x 坐标
y	Integer	Y	0	叠加时自定义的位置，y 坐标
position	String	Y	LeftTop	叠加时的位置
font_size	Integer	Y	0	叠加项的字体大小，0 表示自动-大字体，1 表示自动-中等，2 表示自动-较小，其他值表示具体像素大小
style	String	Y	normal	叠加时的文字样式，normal 表示正常，bold 表示加粗
outline	Integer	Y	0	叠加的外边框
font_color	Integer	Y	1	叠加项的字体颜色，0 表示黑色，1 表示白色，2 表示红色，3 表示蓝色，4 表示绿色，5 表示黄色，6 表示紫色，7 表示青色
outline_color	Integer	Y	1	外边框的颜色，0 表示黑色，1 表示白色，2 表示红色，3 表示蓝色，4 表示绿色，5 表示黄色，6 表示紫色，7 表示青色

enable	Boolean	Y	true	是否开启该叠加项
--------	---------	---	------	----------

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

6.4.13 删除 Osd 素材

* 接口描述

删除单个 OSD 叠加项

* 请求地址

/api/codec/{interface}/vpss/osd/{osd_id}

* 请求方法

DELETE

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

6.4.14 将裁切参数应用到布局设置中

* 接口描述

在画中画模式下请求该接口，可以将 hdmi/sdi 的裁切参数应用到当前的画中画布局中。需要传入当前布局参数。

* 请求地址

/api/codec/{interface}/vpss/input/crop_apply

* 请求方法

POST

* 请求参数

```
[
  {
    "height": 100,
    "width": 100,
    "name": "hdmi",
    "x": 0,
    "y": 0,
    "zIndex": 1,
    "enable": true
  },
  {
    "height": 33.33,
    "width": 33.33,
    "name": "sdi",
    "x": 65,
    "y": 64,
    "zIndex": 2,
    "enable": true
  }
]
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
height	Integer	Y	100	画面的图像高度
width	Integer	Y	100	画面的图像宽度
name	String	Y	hdmi	所在视频源名称, hdmi/sdi
x	Integer	Y	0	画面的 x 坐标, 按百分比计算
y	Integer	Y	0	画面的 y 坐标, 按百分比计算
zIndex	Integer	Y	1	画面层级, 数字越大层级越高
enable	Boolean	Y	true	当前视频源画面是否显示

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

6.4.15 获取静帧图片列表

* 接口描述

获取“静帧代播”的图片。当 HDMI/SDI/USB 没有信号的时候, 该图片将作为默认输出图片

* 请求地址

/api/codec/vpss/static_frame

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
[
  {
    "size_h": 1080,
    "size_w": 1920,
    "url": "/frame/static_frame.png",
    "path": "/data/storage/frame/static_frame.png",
    "yuv_path": "/data/storage/frame/static_frame.yuv",
    "id": "static_frame",
    "pixel_format": 0
  }
]
```

* 响应参数说明

字段名	字段类型	字段描述
id	String	图片唯一 ID。目前固定为"static_frame"
size_h	Integer	图片尺寸高度
size_w	Integer	图片尺寸宽度
url	String	拼接 host 即可获取静态图片资源
path	String	图片的系统路径
yuv_path	String	yuv 文件的系统路径

6.4.16 上传静帧图片

* 接口描述

上传静帧图片

* 请求地址

/api/codec/vpss/static_frame

* 请求方法

POST

* 请求参数

```
{
  "file": (binary)
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
file	Binary	Y		图片文件

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

6.4.17 重置静帧预设配置

* 接口描述

重置静帧图片为默认图片

* 请求地址

/api/codec/vpss/static_frame/reset

* 请求方法

POST

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

7 语音对讲/Service of talk

7.1 语音对讲/Service of talk

7.1.1 获取通话服务版本

* 接口描述

获取语音对讲服务器当前版本号

* 请求地址

/api/talk/talkVersion

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "msg": {
    "version": "1.10.0003"
  }
}
```

* 响应参数说明

字段名	字段类型	字段描述
version	String	版本号

7.1.2 设置语言

* 接口描述

设置语言

* 请求地址

/api/talk/setLanguage

* 请求方法

POST

* 请求参数

```
{
  "language": "cn"
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
language	String	Y	en	语言

* 响应参数

```
{  
  "result": "ok",  
  "msg": null  
}
```

* 响应参数说明

无

7.1.3 查询本机序列号

* 接口描述

查询本机序列号

* 请求地址

/api/talk/queryLocalDeviceSerial

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{  
  "result": "ok",  
  "msg": {  
    "serial": "201412000F1DB"  
  }  
}
```

* 响应参数说明

字段名	字段类型	字段描述
serial	String	序列号

7.1.4 删除设备

* 接口描述

删除设备

* 请求地址

/api/talk/deleteDevice

* 请求方法

POST

* 请求参数

```
{  
  "device_ip": "192.168.43.176"  
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
device_ip	String	Y	192.168.43.176	设备 IP

* 响应参数

```
{  
  "result": "ok",  
  "msg": null  
}
```

* 响应参数说明

无

7.1.5 查询设备列表

* 接口描述

查询设备列表

* 请求地址

/api/talk/queryDeviceList

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{  
  "result": "ok",  
  "msg": {  
    "local_device": {  
      "name": "P3-201412000F1DB",  
      "ip": "192.168.35.35",  
      "type": "device",  
      "usb_status": false,  
      "talk_status": false,  
    }  
  }  
}
```

```

    "connect_status": "noconnect",
    "connect_ip": null,
    "connect_type": null,
    "version": "1.10.0003",
    "new_version": true,
    "http_port": 80,
    "https_port": 443
  },
  "other_device": []
}
}
}

```

* 响应参数说明

字段名	字段类型	字段描述
name	String	设备名
ip	String	设备 IP
type	String	设备类型
usb_status	Boolean	USB 状态
talk_status	Boolean	通话状态
connect_status	String	连接状态
connect_ip	String	连接对象 IP
connect_type	String	连接对象类型
version	String	通话服务版本
new_version	Boolean	新版还是旧版
http_port	Integer	开放的 http 端口
https_port	Integer	开放的 https 端口

7.1.6 添加设备

* 接口描述

添加设备

* 请求地址

/api/talk/addDevice

* 请求方法

POST

* 请求参数

```

{
  "auth_code": "",
  "type": "device",
  "ip": "192.168.40.100",

```

```
"http_port": 80,  
"https_port": 443,  
"serial": "7116110002301"  
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
auth_code	String	Y		KIS 授权码, type 为"device"时 auto_code 为 None
type	String	Y	device	类型, device, server
ip	String	Y	192.168.40.100	设备 ip
http_port	Integer	Y	80	设备开放的 http 端口, 默认为:80
https_port	Integer	Y	443	设备开放的 https 端口, 默认为:443
serial	String	Y	7116110002301	序列号

* 响应参数

```
{  
  "result": "ok",  
  "msg": null  
}
```

* 响应参数说明

无

7.1.7 设置本机设备与设备的连接

* 接口描述

设置本机设备与设备的连接

* 请求地址

/api/talk/setLDLinkDEnable

* 请求方法

POST

* 请求参数

```
{  
  "l_dev_ip": "192.168.38.74",  
  "dev_ip": "192.168.40.100",  
  "enable": true  
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
------	------	------	-------	----

l_dev_ip	String	Y	192.168.38.74	本机设备 IP
dev_ip	String	Y	192.168.40.100	目标 IP
enable	Boolean	Y	true	是否通话

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

7.1.8 设置网页与设备的连接

* 接口描述

设置网页与设备连接

* 请求地址

/api/talk/setWLinkDEnable

* 请求方法

POST

* 请求参数

```
{
  "web_ip": "192.168.40.122",
  "dev_ip": "192.168.43.176",
  "enable": true
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
web_ip	String	Y	192.168.40.122	网页 IP
dev_ip	String	Y	192.168.43.176	设备的 IP
enable	Boolean	Y	true	是否通话

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

7.1.9 设置设备和语音服务器的连接

* 接口描述

设置设备与语音服务器的连接

* 请求地址

/api/talk/setLDLinkSEnable

* 请求方法

POST

* 请求参数

```
{
  "l_dev_ip": "192.168.35.11",
  "ser_ip": "192.168.40.100",
  "enable": true
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
l_dev_ip	String	Y	192.168.35.11	本机设备的 IP
ser_ip	String	Y	192.168.40.100	目标服务器的 IP
enable	Boolean	Y	true	是否通话

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

7.1.10 设备请求连接，设备与本机设备间连接(接口不对前端开放)

* 接口描述

设备请求连接,设备与本机设备间连接(接口不对前端开放)

* 请求地址

/api/talk/dLinkLDEnable

* 请求方法

POST

* 请求参数

```
{
  "c_dev_ip": "192.168.40.10",
  "c_dev_info": {
```

```
"version": "1.10.0001",
"http_port": 80,
"https_port": 443
},
"s_dev_ip": "192.168.40.100"
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
c_dev_ip	String	Y	192.168.40.10	设备 IP
c_dev_info	Object	Y		设备信息
s_dev_ip	String	Y	192.168.40.100	本机设备 IP

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无

7.1.11 通知设备删除通话(接口不对前端开放)

* 接口描述

通知设备删除通话(接口不对前端开放)

* 请求地址

/api/talk/notifyDevDelTalk

* 请求方法

POST

* 请求参数

```
{
  "c_ip": "192.168.40.10",
  "s_ip": "192.168.40.100"
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
c_ip	String	Y	192.168.40.10	客户端设备 IP
s_ip	String	Y	192.168.40.100	服务端设备 IP

* 响应参数

```
{
  "result": "ok",
}
```

```
"msg": null
}
```

* 响应参数说明

无

7.1.12 查询设备连接状态

* 接口描述

查询设备连接状态

* 请求地址

/api/talk/queryLDeviceConnectStatus

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "result": "ok",
  "msg": {
    "connect_status": "noconnect",
    "connect_ip": "0.0.0.0"
  }
}
```

* 响应参数说明

字段名	字段类型	字段描述
connect_status	String	设备连接状态 未连接: noconnect; 连接中: connecting; 已连接: connected
connect_ip	String	设备 ip

8 Onvif 服务/Service of onvif

8.1 Onvif 服务/Service of onvif

8.1.1 获取 onvif 服务状态

* 接口描述

查询设备连接状态

* 请求地址

/api/onvifmanage/onvif/get

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "discoverable": true,
  "control": {
    "enabled": true,
    "discover_port": 3702,
    "port": 8080
  },
  "admin_name": "admin",
  "admin_password": "",
  "auth_req": false,
  "location": "default",
  "device_name": "",
  "device_type": ""
}
```

* 响应参数说明

字段名	字段类型	字段描述
discoverable	Boolean	设备是否能被发现
control	Object	onvif 控制选项
enable	Boolean	onvif 服务是否开启
discover_port	Integer	onvif 发现服务端口
port	Integer	设备发现端口
admin_name	String	用户名
admin_password	String	密码
auth_req	Boolean	是否开启用户鉴权
location	String	位置信息
device_name	String	设备名称
device_type	String	设备类型

8.1.2 设置 onvif 服务状态

* 接口描述

设置 onvif 服务状态

* 请求地址

/api/onvifmanage/onvif/set

* 请求方法

POST

* 请求参数

```
{
  "control": {
    "enabled": true,
    "discover_port": 3702,
    "port": 8080
  },
  "discoverable": true,
  "auth_req": false,
  "admin_name": "Admin",
  "admin_password": "",
  "location": "default",
  "device_name": "",
  "device_type": ""
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
enabled	Boolean	Y	false	onvif 服务是否开启
discover_port	Integer	Y	3702	onvif 发现服务端口（暂不支持修改）
port	Integer	Y	8080	设备发现端口
discoverable	Boolean	Y	true	当 onvif 服务开启时，设备是否能被自动发现
auth_req	Boolean	Y	true	是否开启用户鉴权
admin_name	String	Y	admin	开启鉴权时，用于校验的用户名
admin_password	String	Y	Admin123	开启鉴权时，用于校验的用户密码，若为空则默认为 Admin123
location	String	Y	default	位置信息
device_name	String	Y	E3	设备名称
device_type	String	Y	E3	设备类型

* 响应参数

```
{
  "result": "ok",
}
```

```
"msg": null
}
```

* 响应参数说明

无

9 NDI bridge 服务/Bridge service

9.1 NDI bridge 服务/Bridge service

9.1.1 获取 bridge 设置

* 接口描述

调用此接口以获取 ndi bridge 当前配置。

* 请求地址

/api/bridge/get

* 请求方法

GET

* 请求参数

无

* 请求参数说明

无

* 响应参数

```
{
  "enable": true,
  "recv": "Public",
  "address": "192.168.35.70",
  "port": "5990",
  "secure": "",
  "id": "E3"
}
```

* 响应参数说明

字段名	字段类型	字段描述
enable	Boolean	ndi bridge 是否开启
recv	String	要共享到接入组的组名称，默认为 Public
address	String	要连接的 NDI Host 的公网 IP
port	String	NDI Host 的对应端口号，默认 5990
secure	String	连接所需的加密密钥，默认为空
id	String	映射到 NDI Host 的源的名称

9.1.2 设置 bridge 配置

* 接口描述

设置 bridge 配置。

* 请求地址

/api/bridge/set

* 请求方法

POST

* 请求参数

```
{
  "enable": true,
  "recv": "Public",
  "address": "192.168.35.70",
  "port": "5990",
  "secure": "",
  "id": "E3"
}
```

* 请求参数说明

参数名称	参数类型	是否必填	参数值样例	描述
enabled	Boolean	Y	false	ndi bridge 是否开启
recv	String	Y	Public	要共享到接入组的组名称，默认为 Public
address	String	Y	192.168.35.70	要连接的 NDI Host 的公网 IP
port	String	Y	5990	NDI Host 的对应端口号，默认 5990
secure	String	Y		连接所需的加密密钥，默认为空
id	String	Y	E3	映射到 NDI Host 的源的名称

* 响应参数

```
{
  "result": "ok",
  "msg": null
}
```

* 响应参数说明

无