

linkingvision



API 手册

Copyright © 2020 All rights reserved

版本记录

版本	日期	描述
1.00	2018/04/03	第一个版本
1.01	2018/04/25	r4.2
1.02	2018/05/05	r5.0
1.03	2018/07/14	r6.0 拆分 API 手册
1.04	2019/04/09	r8.0 加入设备 SDK API
2.00	2020/01/15	API 格式合进发布版本

内容

1.0 发布说明	5
1.1 版本 1.00	5
2.0 范围	5
3.0 参考链接	5
4.0 常用术语	6
5.0 HTTP/HTTPS RESTFUL接口	7
5.1 协议格式	7
5.2 API文档	7
5.3 API Postman工程	8
5.4 API 认证机制	10
6.0 API接口	11
6.1 系统API	11
6.2 源管理API	12
6.3 设备管理API	13
6.4 国标GB管理API	14
6.5 云级联管理API	15
6.6 ONVIF API	16
6.7 监控点管理 API	17
6.8 录像管理 API	18
6.9 视频会议管理 API	19
7.0 JS API 接口	20
7.1 JS 对象参数	20
7.2 Websocket	21
7.3 WebRTC	21
7.4 HLS	22
7.5 RTMP	22
7.6 视频会议	22

1.0 发布说明

1.1 版本 1.00

初始版本.

2.0 范围

文档包含 h5stream 互联网直播方案的 API 定义。

3.0 参考链接

ONVIF <https://www.onvif.org/>

4.0 常用术语

源: RTSP RTMP ONVIF 等单通道视频源, 或者是添加设备后自动产生的通道也称为源

设备: 摄像机 NVR 等通过各厂家设备 SDK 添加设备, 添加设备后会产生一个或多个源, 源对应不同的通道。

5.0 HTTP/HTTPS RESTFUL 接口

RESTFUL 接口是基于 HTTP 的简单的无状态的协议，所有的请求采取 GET 方法，HTTP 返回都采用 JSON 格式。调试的时候可以使用 HTTP，在实际的生产环境中建议使用 HTTPS。为了方便后续的例子都是用 HTTP，如果换 HTTPS，只需要把对应的 http 换成 https 即可。

5.1 协议格式

所有的请求都采用 HTTP GET，返回采用 JSON 格式，具体返回的 JSON 格式根据不同的命令而不同，下表是基本的格式和例子，为了让大家更容易理解，有些命令介绍中直接使用代表性的值。

格式:

```
Request : http://server/api/v1/<resources>?para=xxx
         &<...>=<...>
Response:
{
  "bStatus": true/false,
  "strCode": "xxxxxxx"
}
\r\n
```

<... >=<... >代表多个请求参数.

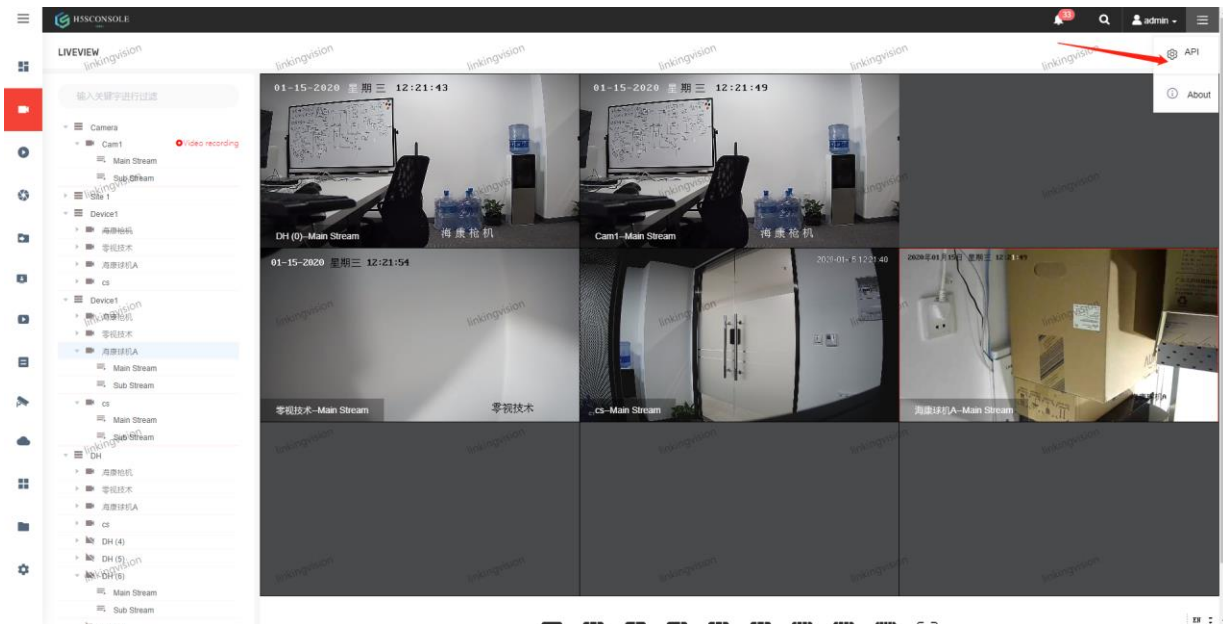
例子:

```
Request :
http://localhost:8080/api/v1/AddSrcONVIF?name=name1&token=token4&user=admin&password=12345&ip=192.168.0.234&port=80&onvifaddr=/onvif/device_service&session=c1782caf-b670-42d8-ba90-2244d0b0ee83
Response:
{
  "bStatus": true,
  "strCode": "Add successfully"
}
\r\n
```

5.2 API 文档

参考 H5S 视频平台中文手册安装后，可以访问 H5SCONSOLE 管理界面

<http://192.168.100.127:8080/> 访问后可以点击 API 菜单, 如下图所示:



点击 API 菜单后会出现 H5S 视频平台 API 文档, 如下图所示

H5S视频平台 r10-API文档 关于零视

搜索接口

接口总数: 11 最后更新时间: 2020-01-14 11:35:42

接口	状态码	接口名称	接口URI	所在分组	更新日期
分组		★ 接口名称	接口URI	所在分组	更新日期
所有接口		GetCodecInfo	GET http://192.168.100.127:8080/api/v1/GetCodecInfo?session=3db6e64e-eb7f-413d-9f...	01 系统(System)	2020-01-12 11:04:51
01 系统(System)		GetDeviceSummary	GET http://192.168.100.127:8080/api/v1/GetDeviceSummary?session=3db6e64e-eb7f-4...	01 系统(System)	2020-01-12 11:02:58
02 设备管理(Device Manager)		GetRunInfo	GET http://192.168.100.127:8080/api/v1/GetRunInfo?session=3db6e64e-eb7f-413d-9f...	01 系统(System)	2020-01-12 11:02:46
03 设备管理(Device Manager)		GetSystemInfo	GET http://192.168.100.127:8080/api/v1/GetSystemInfo?session=3db6e64e-eb7f-413d...	01 系统(System)	2020-01-12 11:58:46
04 设备管理(Device Manager)		GetVolumes	GET http://192.168.100.127:8080/api/v1/GetVolumes?session=3db6e64e-eb7f-413d-9f...	01 系统(System)	2020-01-12 10:57:15
05 设备管理(Device Manager)		h5srtcapl Status	GET http://192.168.100.127:8080/api/v1/h5srtcapl/status?session=3db6e64e-eb7f-413d...	01 系统(System)	2020-01-13 16:35:56
06 ONVIF		h5swsapi Status	GET http://192.168.100.127:8080/api/v1/h5swsapi/Status?session=3db6e64e-eb7f-413d...	01 系统(System)	2020-01-13 16:35:34
07 录像回放(Camera)		Keepalive	GET http://192.168.100.127:8080/api/v1/Keepalive?session=3db6e64e-eb7f-413d-9f...	01 系统(System)	2020-01-12 11:03:54
08 Record(录像)		Login	GET http://192.168.100.127:8080/api/v1/Login?user=admin&password=827ccb0eea8a7...	01 系统(System)	2020-01-12 11:59:02
09 视频会议(Conference)		Logout	GET http://192.168.100.127:8080/api/v1/Logout?session=3db6e64e-eb7f-413d-9f...	01 系统(System)	2020-01-12 11:59:15
10 集群(Cluster)		UpdateUser	GET http://192.168.100.127:8080/api/v1/UpdateUser?user=admin&oldpassword=827cc...	01 系统(System)	2020-01-12 12:00:21
99 内部Internal					

5.3 API Postman 工程

H5S 视频平台 API 提供了 Postman 工程, 可以按如下方法导入工程并开始 API 测试
在导入工程的之前请先安装好 Postman。

下载如下三个文件

H5S VIDEO PLATFORM r10.postman_collection.json

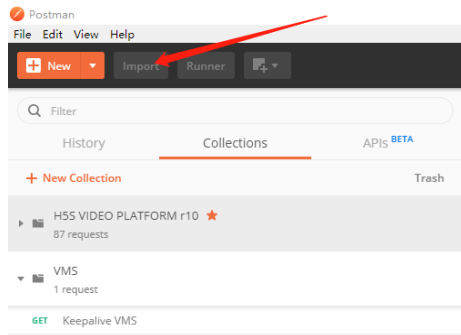
H5STREAM ENV HTTP.postman_environment.json
H5STREAM ENV HTTPS.postman_environment.json

可以从如下链接中下载

<https://linkingvision.cn/download/documentation/h5stream/postman/>

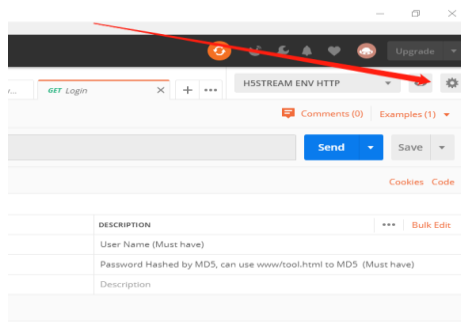
5.3.1 导入工程

参考下图导入 H5S VIDEO PLATFORM r10.postman_collection.json 文件

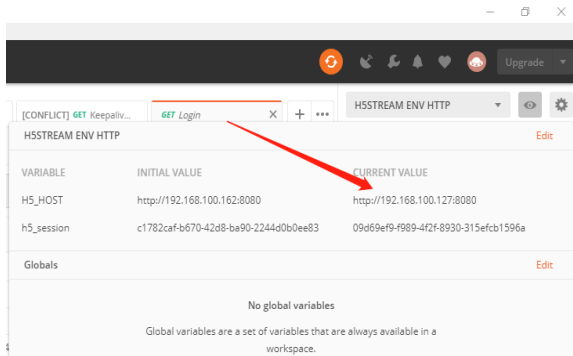


5.3.2 导入HTTP/HTTPS 环境

参考下图导入 H5STREAM ENV HTTP.postman_environment.json 和 H5STREAM ENV HTTPS.postman_environment.json

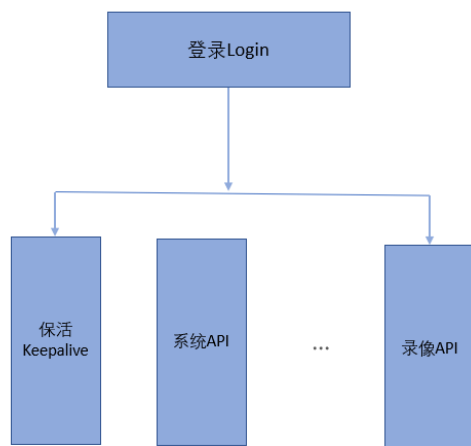


导入后可以参考下图修改对应 H5STREAM 服务的 IP 地址和端口



5.4 API 认证机制

视频平台采用基于 token 的 session 管理机制，如果使用 Login API 获取 session，然后所有的 API 调用都使用该 session，session 是有超时时间的，可以使用 Keepalive 保活该 session，这样 session 就不会超时，session 默认的超时时间是 600 秒。



6.0 API 接口

6.1 系统 API

名称	描述
Login	登录
Keepalive	保活
UpdateUser	更新用户
Logout	登出
GetSystemInfo	获取系统信息
GetCodeclInfo	获取编解码信息
GetRunInfo	获取运行信息
GetDeviceSummary	获取设备接入汇总信息
GetVolumes	获取磁盘列表
h5swsapi/Status	获取 WS 连接信息
h5srtcapi/Status	获取 RTC 连接信息

6.2 源管理 API

名称	描述
GetSrc	获取源列表
GetSrcWithoutDevice	获取除了设备 SDK 产生的源列表
GetSrcCamera	获取 RTSP RTMP ONVIF 源列表
GetImage	获取源最近缓存的图片
AddSrcFile	添加文件模拟源
AddSrcRTSP	添加 RTSP RTMP 源
AddRTMPPush	添加 RTMP 推流源
AddSrcONVIF	添加 ONVIF 源
DelSrc	删除源
Ptz	云台操作
SetPreset	设置预置点
DelPreset	删除预置点
GetPresets	获取预置点列表

6.3 设备管理 API

名称	描述
GetDevice	获取设备列表
GetDeviceSrc	获取设备源列表
AddDeviceHik	添加海康 NVR SDK 设备
AddDeviceDh	添加大华 NVR SDK 设备
AddDeviceHikISC	添加海康 ISC 平台
AddDeviceTd	添加天地伟业 NVR SDK 设备

6.4 国标 GB 管理 API

名称	描述
GetGbDevice	获取注册国标设备列表
RebootGbDevice	重启国标设备
RefreshGbDevice	刷新国标设备, 用于刷新国标设备通道
GetGbDeviceSrc	获取国标设备下的源列表
GetGbPlatform	获取上联国标平台列表
AddGbPlatform	添加国标上联平台
DelGbPlatform	删除国标上联平台

6.5 云级联管理 API

名称	描述
GetCloudClientInfo	获取级联客户端配置
GetCloudDevice	获取级联注册设备列表
GetCloudDeviceSrc	获取级联注册

6.6 ONVIF API

名称	描述
OnvifSearch	搜索 ONVIF设备
OnvifProbe	探测 ONVIF 设备

6.7 监控点管理 API

名称	描述
GetCamera	获取监控点配置列表
AddCamera	添加监控点配置
DelCamera	删除监控点配置

6.8 录像管理 API

名称	描述
Snapshot	服务端抓图
PbSnapshot	服务端回放抓图
ManualRecordStart	手动录像开始
ManualRecordStop	手动录像结束
Search	录像 抓图 归档搜索
SearchByFilename	根据文件名搜索录像
Archive	归档设备录像
GetArchiveStatus	获取归档状态
SearchDeviceRecordByTime	根据时间搜索设备录像
SearchDeviceRecordCalendar	搜索设备录像日历(仅支持海康 SDK 接入)

6.9 视频会议管理 API

名称	描述
h5sconference/User	获取加入会议用户列表

7.0 JS API 接口

7.1 JS 对象参数

```
/**
 * Interface with h5s websocket player API
 * @constructor
 * @param
 var pbconf1 = {
     begintime: '2019-03-23T120101+08',//{string} begintime 0 for fileplayback
     endtime: '2019-03-23T150101+08',//{string} endtime 0 for fileplayback
     autoplay: 'true', // 'true' or 'false' for playback autoplay
     showposter: 'true', //'true' or 'false' show poster
     serverpb: 'true', //'true' or 'false' playback from h5stream record, default false
     filename: 'token1.mp4', // file name need to playback (begintime == 0 & endtime == 0 and serverpb is true)
     callback: PlaybackCB, //{function}(event(string), userdata(object))
     userdata: user data // user data
 };

 var conf = {
     videoid:'h5sVideo1', //{string} - id of the video element tag
     videodom: h5svideodom1, //{object} - video dom. if there has videoid, just use the videoid
     protocol: window.location.protocol, // {string} - 'http:' or 'https:'
     host: window.location.host, //{string} - 'localhost:8080'
     rootpath:window.location.pathname, // {string} - path of the app running
     token:'token1', // {string} - token of stream
     streamprofile: 'main', // {string} - stream profile, main/sub or other predefine transcoding profile
     pbconf: pbconf1, //This is optional, if no pbconf, this will be live.
     hlsver:'v1', //{string} - v1 is for ts, v2 is for fmp4
     session:'c1782caf-b670-42d8-ba90-2244d0b0ee83', //{string} - session got from login
     consolelog: 'true' // 'true' or 'false' enable/disable console.log
 };
 */
```

当 pbconf 不存在的时候表示实时视频。

当 serverpb 为 false 或者不存在的时候表示从 NVR 中回放，从 NVR 中回放都采用按时间回放，当 serverpb 为 true 表示从 h5s 的录像记录中回放，从 h5s 回放有两种方式：按时间回放和按文件名回放，当 begintime 和 endtime 都为 '0' 时表示按文件名回放，按文件名回放的时候 filename 为录像指令返回的文件名，比如"strFileName": "107-e88c9fae-2c56-4329-b9e2-ec99dd36ab9d.mp4"。

NVR 回放可以参考 www/playback2.html

H5S 基于文件的回放可以参考 www/serverfilepb.html

H5S 基于时间的回放可以参考 www/serverpb.html

consolelog 可以设置为 false，在 Chrome 的调试窗口就不会打印 h5splayer.js 的日志了。

7.2 Websocket

```
/**
 * Interface with h5s websocket player API
 * @constructor
 */
function H5sPlayerWS(conf)
H5sPlayerWS.prototype.connect
H5sPlayerWS.prototype.disconnect
```

回放 API

```
H5sPlayerWS.prototype.start
H5sPlayerWS.prototype.pause
H5sPlayerWS.prototype.resume
H5sPlayerWS.prototype.speed(x) //x 0.5 1 2 4 8 16
H5sPlayerWS.prototype.seek(x) //x 相对开始回放时间的偏移 单位为秒
```

WS 视频播放可以参考 www/ws.html

7.3 WebRTC

```
/**
 * Interface with h5s WebRTC player API
 * @constructor
 */
function H5sPlayerRTC(conf)
H5sPlayerRTC.prototype.connect
H5sPlayerRTC.prototype.disconnect
```

回放 API

```
H5sPlayerWS.prototype.start
H5sPlayerWS.prototype.pause
H5sPlayerWS.prototype.resume
H5sPlayerWS.prototype.speed(x) //x 0.5 1 2 4 8 16
H5sPlayerWS.prototype.seek(x) //x 相对开始回放时间的偏移 单位为秒
```

WS 视频播放可以参考 www/rtc.html

7.4 HLS

```
/**
 * Interface with h5s websocket player API
 * @constructor
 */
function H5sPlayerHls(conf)
H5sPlayerHls.prototype.connect
H5sPlayerHls.prototype.disconnect
```

WS 视频播放可以参考 www/hls.html

7.5 RTMP

为了支持老版本的一些浏览器，H5Stream 支持基于 Flash 的 RTMP 视频播放采用的播放器是 videojs，具体使用方法可以参考 www/rtmp.html，如果需要支持双击全屏功能，可以参考 www/rtmp2.html。

7.6 视频会议

```

/**
 * Interface with h5s WebRTC Conference API
 * @constructor
 * @param
var conf = {
    localvideoid:'h5sVideoLocal', //{string} - id of the local video element tag
    remotevideoid:'h5sVideoRemote', //{string} - id of the remote video element tag
    //localvideodom: h5svideodomlocal, //{object} - local video dom. if there has videoid, just use the
videoid
    //remotevideodom: h5svideodomremove, //{object} - remote video dom. if there has videoid, just use
the videoid
    protocol: window.location.protocol, //http: or https:
    host: window.location.host, //localhost:8080
    rootpath:'', // {string} - path of the app running
    callback: EventCB, //Callback for the event
    userdata: null, // user data
    name: 'manager', //name of the user
    session: strSession //session got from login
};
*/

/**
 * Interface with h5s WebRTC Conference API
 * @constructor
 */
function H5sConference(conf)
H5sConference.prototype.connect
H5sConference.prototype.disconnect
H5sConference.prototype.call
H5sConference.prototype.answer
H5sConference.prototype.hangup

```

其中 callback 回调会有三种消息

CFE_EVENT_PEER_CALL(strId 一个参数, 参会用户的 ID)

CFE_EVENT_PEER_ADD(strId 和 strName 两个参数, 分别为参会用户的 ID 和名称)

CFE_EVENT_PEER_DEL(strId 一个参数, 参会用户的 ID)

视频会议例子可以参考 www/conference.html

视频会议需要使用 HTTPS 访问, 类似这样的 URL <https://192.168.100.127:8443/conference.html>