linkingvision

H5STREAM

User Manual

Revision History

Revision	Date	Description
1.00	2018/04/03	First version
1.01	2018/04/26	r4.2 update

Table of Contents

1.0 Release Notes

1.1 Version 1.00

Initial version.

2.0 Scope

This specification defines live streaming scenario, and include install guide, development API interface. And introduce the best live streaming method for different browser, and then can achieve sub seconds low latency.

3.0 References

ONVIF https://www.onvif.org/

4.0 Terms and Definitions

5.0 Specification overview

Recently, as the Internet also the Mobile grow quickly, more and more Mobile APP based on HTML5, so the big challenge is the video streaming for browser, and Flash has a lot of problem, so most Brower has begin to stop flash by 2020. And all the browser has top the plugin. Currently IP Camera/NVR/VMS web browser only support IE11, so it is import to support video streaming in browser without plugin.

And the HTML5 native video streaming technology are very different. And the low latency is an important requirement, so how to achieve 1s or 500ms latency is a big challenge.

The cloud is very common now, remote cloud video streaming and video sharing with low latency is also important.

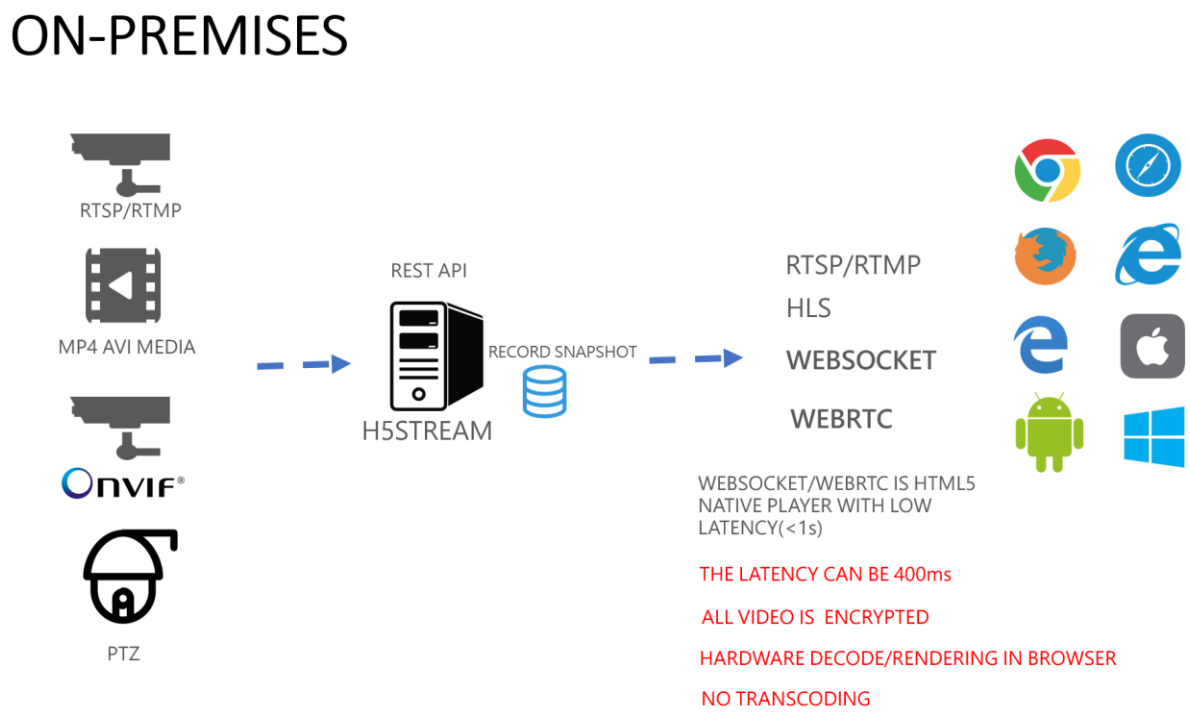
H5STREAM support the HTML5 native video streaming and cloud video streaming through WEBRTB WEBSOCKET RTMP FLV. Below is all the technology we used for different browser.

	Chrome	Firefox	IE11	Edge	Safari	WeChat
WIN7	WEBRTC WEBSOCKET	WEBRTC WEBSOCKET	RTMP	-	-	-
WIN 8/10	WEBRTC WEBSOCKET	WEBRTC WEBSOCKET	RTMP WEBSCOKET	WEBRTC	-	-
macOS	WEBRTC WEBSOCKET	WEBRTC WEBSOCKET	-	-	WEBRTC WEBSOCKET	-
iOS 11	HLS WEBRTC	HLS WEBRTC	-	-	HLS WEBRTC	HLS WEBRTC
iOS 8-10	HLS	HLS	-	-	HLS	HLS
Android	WEBSOCKET WEBRTC	WEBRTC WEBSOCKET	-	-	-	HLS

Browser Compatibility

6.0 On premises

H5STREAM is streaming platform which support Windows Linux(CentOS ubuntu) macOS.



On Premises

6.1 Video source support

H5STREAM support use MP4/AVI file as video source, so it is very convenient to test without a real video source. Because most IP camera support RTSP, And old streaming system only support RTMP. So H5STREAM both support RTSP and RTMP. As a video surveillance standard, ONVIF has been supported in most of the video surveillance device and system, so H5STREAM support ONVIF directly. And user can use H5STREAM RESTFUL API to control ONVIF PTZ device.

6.2 Platform

H5STREAM is a cross platform video streaming including Windows 7/8/10， CentOS ubuntu macOS, and you also can run H5STREAM in Azure or AWS. And also you can run as cluster based on NGINX.

6.3 Protocol

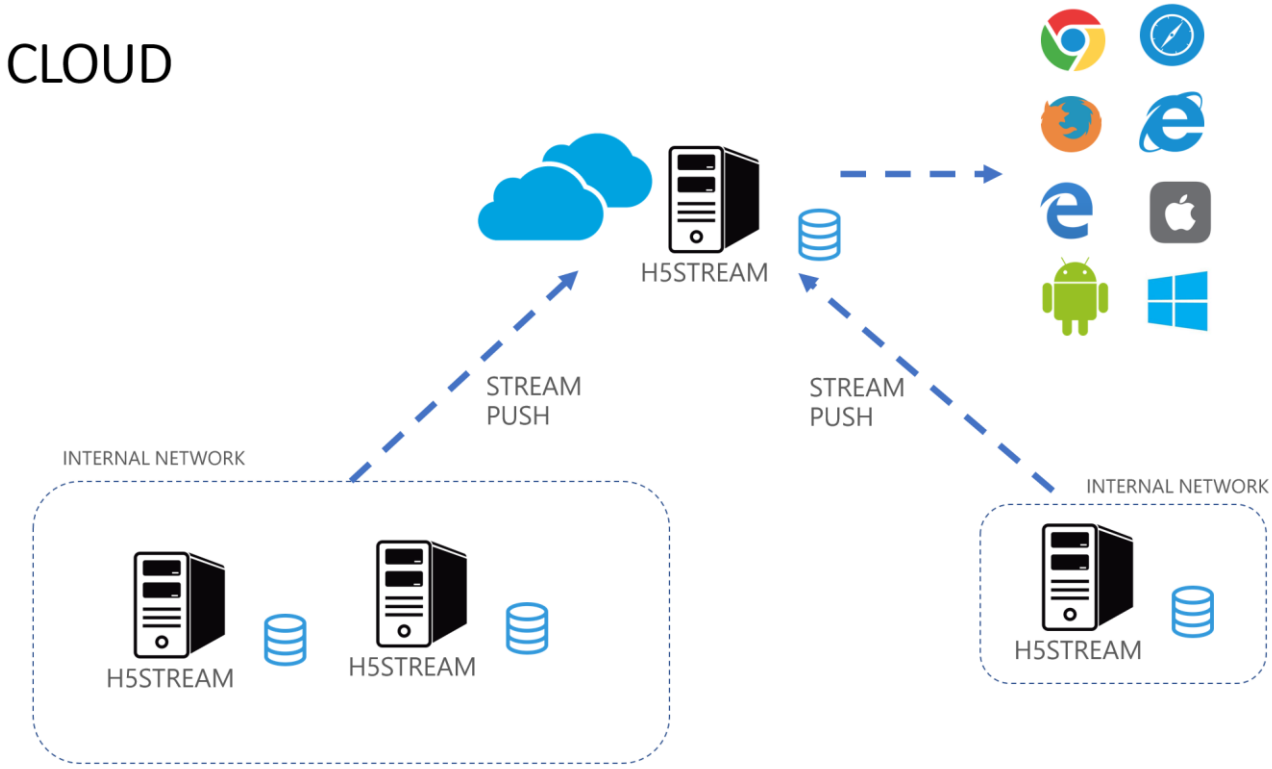
RTMP/RTSP/HLS is the very common video streaming protocol, H5STREAM support all of them, and for new WEBSOCKET and WEBRTC, H5STREAM support them very well, so can use the hardware decoding and then achieve low latency.

6.4 Video encryption

H5STREAM use native HTML5 video streaming, so all the video streaming will based on HTTPS or SSL/TLS for UDP.

7.0 Cloud

When user need access remote video, and maybe you need map NAT port or use DDNS, it is not easy to do this for end user. H5STREAM support video push mode, and all the video is encrypted, so make the communication safe.



Cloud video Streaming

8.0 Installing Software

8.1 Prepare

Windows 7/8/10 Centos ubuntu macOS 32bit (Only support Windows 32bit)/64bit system, 4G memory, 1 core CPU or above.

Download the package

Download the package you need from below link.

https://www.linkingvision.com/download/

Install vs2017 redistributable

32bit

https://download.visualstudio.microsoft.com/download/pr/100349138/88b50ce70017bf10f2d56d60fcba6ab1/VC_redist.x86.exe

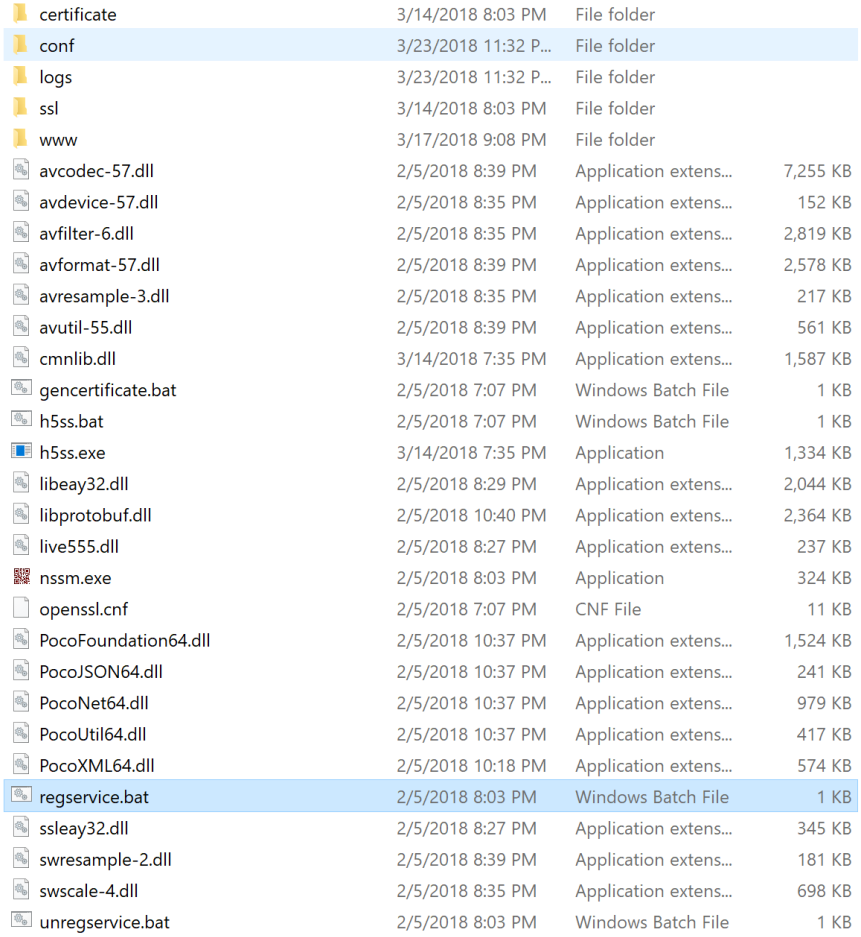
64bit

https://download.visualstudio.microsoft.com/download/pr/11100230/15ccb3f02745c7b206ad10373cbca89b/VC_redist.x64.exe

8.2 Install

Manual running

exec the h5ss.bat, or h5ss.sh.



certificate	3/14/2018 8:03 PM	File folder	
conf	3/23/2018 11:32 P...	File folder	
logs	3/23/2018 11:32 P...	File folder	
ssl	3/14/2018 8:03 PM	File folder	
www	3/17/2018 9:08 PM	File folder	
avcodec-57.dll	2/5/2018 8:39 PM	Application extens...	7,255 KB
avdevice-57.dll	2/5/2018 8:35 PM	Application extens...	152 KB
avfilter-6.dll	2/5/2018 8:35 PM	Application extens...	2,819 KB
avformat-57.dll	2/5/2018 8:39 PM	Application extens...	2,578 KB
avresample-3.dll	2/5/2018 8:35 PM	Application extens...	217 KB
avutil-55.dll	2/5/2018 8:39 PM	Application extens...	561 KB
cmnlib.dll	3/14/2018 7:35 PM	Application extens...	1,587 KB
gencertificate.bat	2/5/2018 7:07 PM	Windows Batch File	1 KB
h5ss.bat	2/5/2018 7:07 PM	Windows Batch File	1 KB
h5ss.exe	3/14/2018 7:35 PM	Application	1,334 KB
libeay32.dll	2/5/2018 8:29 PM	Application extens...	2,044 KB
libprotobuf.dll	2/5/2018 10:40 PM	Application extens...	2,364 KB
live555.dll	2/5/2018 8:27 PM	Application extens...	237 KB
nssm.exe	2/5/2018 8:03 PM	Application	324 KB
openssl.cnf	2/5/2018 7:07 PM	CNF File	11 KB
PocoFoundation64.dll	2/5/2018 10:37 PM	Application extens...	1,524 KB
PocoJSON64.dll	2/5/2018 10:37 PM	Application extens...	241 KB
PocoNet64.dll	2/5/2018 10:37 PM	Application extens...	979 KB
PocoUtil64.dll	2/5/2018 10:37 PM	Application extens...	417 KB
PocoXML64.dll	2/5/2018 10:18 PM	Application extens...	574 KB
regservice.bat	2/5/2018 8:03 PM	Windows Batch File	1 KB
ssleay32.dll	2/5/2018 8:27 PM	Application extens...	345 KB
swresample-2.dll	2/5/2018 8:39 PM	Application extens...	181 KB
swscale-4.dll	2/5/2018 8:35 PM	Application extens...	698 KB
unregservice.bat	2/5/2018 8:03 PM	Windows Batch File	1 KB

Install As service

Windows

regservice.bat install service.

unregservice.bat remove the service.

Centos 7

```
/opt/h5ss/
|-- certificate
|-- conf
|-- gencertificate.sh
|-- h5ss
|-- h5ss.service
|-- h5ss.service.sh
|-- h5ss.sh
|-- lib
|-- logs
|-- openssl
|-- openssl.cnf
`-- www
```

Copy the release package to /opt/h5ss

cp h5ss.service /usr/lib/systemd/system/

systemctl enable h5ss.service

systemctl start h5ss.service

Ubuntu 16.04

sudo mkdir -p /usr/lib/systemd/system/

sudo apt install system

Copy the release package to /opt/h5ss

cp h5ss.service /usr/lib/systemd/system/

systemctl enable h5ss.service

systemctl start h5ss.service

8.3 Install license

In logs/h5sslog.log and get Hostid， and then send the HostId to info@linkingvision.com , after receive the h5ss.lic license file， copy the h5ss.lic file to conf folder, and restart the h5ss.

9.0 Configure video source

In the release package there has one config file conf/h5ss.conf, you can change the file to add video source, and also you can use RESTFUL API to modify the video source. Below table list all the config item H5STREAM have.

CONFIGURATION FILE

conf/h5ss.conf	
HTTP	HTTP HTTPS server configuration
RTSP	RTSP server configuration, SSL is RTSP over TCP/TLS
RTMP	RTMP server configuration, SSL is RTMP over TCP/TLS
FLV	FLV server configuration, SSL is FLV over HTTPS
HLS	HLS version and parameter configuration
WEBRTC	WEBRTC configuration
SYSTEM	H5stream system configuration such as log and HTTP server thread
USER	User management configuration
SOURCE	Video source configuration, include File/RTSP/RTMP/ONVIF

Configuration file

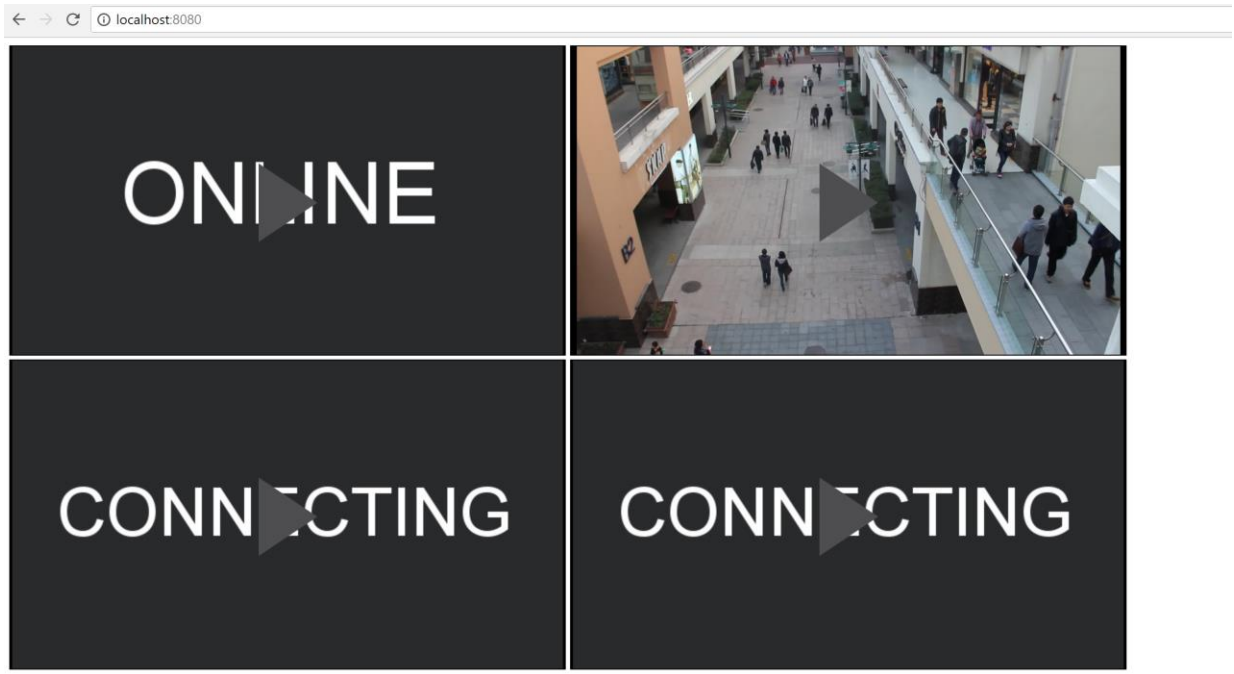
9.1 File source

In the video source config, strToken is the unique for the source, please keep then different. And in the config file nType is H_FILE, and strUrl is video file path, and linkingvision has testing video source for you use, you can download from here https://linkingvision.com/download/H5Stream/video/h5ssample.mp4.

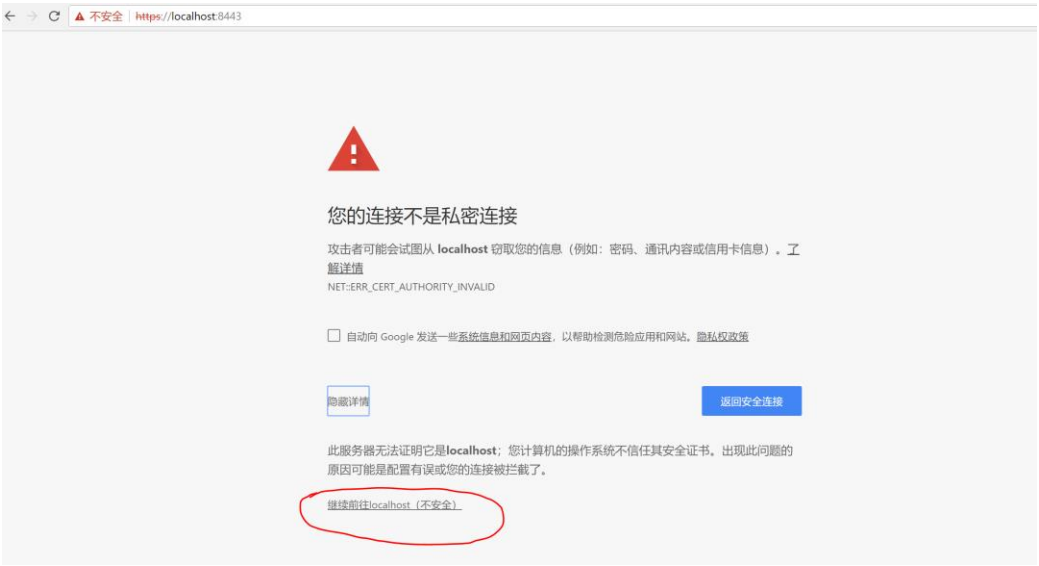
```
"strNameComment": "name for this stream",
"strName": "Stream 1",
"strTokenComment": "token for this stream, must unique, if same, only first will be available",
"strToken": "token1",
"nTypeComment": "source type H5_FILE/H5_STREAM/H5_ONVIF",
"nType": "H5_FILE",
"strUrlComment": "",
"strUrl": "c:\h5ssample.mp4",
"strUserComment": "username",
"strUser": "admin",
"strPasswdComment": "password",
"strPasswd": "12345",
"bPasswdEncryptComment": "Password Encrypted",
"bPasswdEncrypt": false,
"bEnableAudioComment": "Enable Audio",
"bEnableAudio": false
```

File source configuration

After change and save the config file， restart h5ss.bat, In Chrome in put http://localhost:8080/ or https://localhost:8443/, as the https server is self signed certificate, so need click and then go to site.



HTTP Video



HTTPS Video(There has some Chinese, I think you can find the button 😊)

9.2 RTSP RTMP video source

Change nType to H5_STREAM, strUrl is the RTSP RTMP， is the RTSP source need user and password, input user and password to strUser and strPasswd, the user and password in the RTSP URL is not valid.

```
"strNameComment": "name for this stream",
"strName": "Stream 1",
"strTokenComment": "token for this stream, must unique, if same, only first will be available",
"strToken": "token1",
"nTypeComment": "source type H5_FILE/H5_STREAM/H5_ONVIF",
"nType": "H5_STREAM",
"strUrlComment": "",
"strUrl": "rtsp://192.168.0.1/stream",
"strUserComment": "username",
"strUser": "admin",
"strPasswdComment": "password",
"strPasswd": "12345",
```

RTSP/RTMP video source

9.3 ONVIF source

ONVIF has a lot option need config, but most can be default, you can only change the item marked in below picture including nType strUser strPasswd strSrcIPAddress strSrcPort.

```
{
 "strNameComment": "name for this stream",
 "strName": "Stream 1",
 "strTokenComment": "token for this stream, must unique, if same, only first will be available",
 "strToken": "token1",
 "nTypeComment": "source type H5_FILE/H5_STREAM/H5_ONVIF",
 "nType": "H5_ONVIF",
 "strUrlComment": "",
 "strUrl": "rtsp://192.168.0.1/stream",
 "strUserComment": "username",
 "strUser": "admin",
 "strPasswdComment": "password",
 "strPasswd": "12345",
 "bPasswdEncryptComment": "Password Encrypted",
 "bPasswdEncrypt": false,
 "bEnableAudioComment": "Enable Audio",
 "bEnableAudio": false,
 "nConnectTypeComment": "H5_ONDEMAND/H5_ALWAYS/H5_AUTO",
 "nConnectType": "H5_AUTO",
 "nRTSPTypeComment": "RTSP Connect protocol H5_RTSP_TCP/H5_RTSP_UDP/H5_RTSP_HTTP/H5_RTSP_HTTPS/H5_RTSP_AUTO",
 "nRTSPType": "H5_RTSP_AUTO",
 "strSrcIpAddressComment": "Ip Address for the device",
 "strSrcIpAddress": "192.168.0.1",
 "strSrcPortComment": "Port for the device",
 "strSrcPort": "80",
 "nChannelNumberComment": "Channel number (1-512)",
 "nChannelNumber": 1,
 "bOnvifProfileAutoComment": "ONVIF Auto select the video profile",
 "bOnvifProfileAuto": true,
 "strOnvifAddrComment": "",
 "strOnvifAddr": "/onvif/device_service",
 "strOnvifProfileMainComment": "ONVIF Main stream profile name",
 "strOnvifProfileMain": "Profile_1",
 "strOnvifProfileSubComment": "ONVIF Sub stream profile name",
 "strOnvifProfileSub": "Profile_2"
},
```

ONVIF source

10.0 Cloud push mode configuration

10.1 Internal network H5STREAM config

In the cloud part of the configuration, if you want enable cloud push mode, you set the bEnable to true, and config the port. If the cloud H5STREAM use the default config, just input the strCloudIp is OK, you also can use Domain name of the cloud.

```
,,
"cloud": {
 "strServerNameComment": "Server name",
 "strServerName": "Server 1",
 "strServerTokenComment": "Server token",
 "strServerToken": "servertoken1",
 "bEnableComment": "Enable connect",
 "bEnable": false,
 "strCloudIpComment": "Cloud ip address or domain name",
 "strCloudIp": "10.0.0.1",
 "strCloudPortComment": "Cloud port",
 "strCloudPort": "8080",
 "bSSLComment": "Enable SSL for cloud connect",
 "bSSL": false,
 "strUserComment": "User for cloud connect",
 "strUser": "admin",
 "strPasswdComment": "Password MD5 hashed, default 12345",
 "strPasswd": "827ccb0eea8a706c4c34a16891f84e7b"
},
```

11.0 HTTP/HTTPS RESTFUL API

RESTFUL API is based HTTP, all the API use HTTP GET method, and all the return of the API is JSON format. When develop or debug, suggest use HTTP, For the production, suggest use the HTTPS instead of HTTP.

11.1 Protocol Syntax

All HTTP request is based on GET, and all the response is JOSN. And here is the basic syntax and example, for better understanding, some command will use the real value.

Syntax:

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

<… >=<… >stand for multi parameter.

Example:

```
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
```

11.2 System

11.2.1 Login

URL: /api/v1/Login

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/Login?user=xxx&password=xxx
Response:
{
 "bStatus": true,
 "strSession": "c1782caf-b670-42d8-ba90-2244d0b0ee83",
 "nTimeout": 600
}
or
{
 "bStatus": false,
 "strCode": "xxxxxx"
}
\r\n
```

Parameter	Optional/Must	Description
user	*must*	*User name default is admin*
password	*must*	*pass default is 12345, the password is md5 hash of the real password.*

11.2.2 Logout

URL: /api/v1/Logout

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/Logout?session=xxxxxxxxxxxxxxx
Response:
{
 "bStatus": true/false,
 "strCode": "xxxxxx"
}
\r\n
```

Parameter	Optional/Must	Description
session	*must*	*session id*

11.2.3 Keepalive

URL: /api/v1/Keepalive

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/Keepalive?session=xxxxxxxxxxxxxxx
Response:
{
 "bStatus": true/false,
 "strCode": "xxxxxx"
}
\r\n
```

Parameter	Optional/Must	Description
session	*must*	*session id*

11.2.4 Get system info

URL: /api/v1/GetSystemInfo

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/GetSystemInfo?session=xxxxxxxxxxxxxxx
Response:
{
    "strVersion": "r4.0.0403.18",
    "strHostId": "xxxxxxxxx",
    "strLicenseType": "None",
    "strLicenseFull": "",
    "strChannelLimit": "unlimited",
    "strEndtime": "unlimited"
}
\r\n
```

Parameter	Optional/Must	Description
session	*must*	*session id*

11.2.5 Get run info

URL: /api/v1/GetRunInfo

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/GetRunInfo?session=xxxxxxxxxxxxxxx
Response:
{
    "strRunTime": "0H 23MIN",
    "strCPU": "16%",
    "strMemory": "39%",
    "strNetworkIn": "0kpbs",
    "strNetworkOut": "0kpbs"
}\r\n
```

Parameter	Optional/Must	Description
session	*must*	*session id*

11.2.6 Update User

URL: /api/v1/UpdateUser

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/
UpdateUser?user=admin&oldpassword=827ccb0eea8a706c4c34a16891f84e7b&newpa
ssword=xxxxxx&session=xxxxxx
Response:
{
    "bStatus": true,
    "strCode": "Update user successfully"
}\r\n
```

Parameter	Optional/Must	Description
session	*must*	*session id*
user	*must*	*User name*
oldpassword	*must*	*Old password*
newpassword	*must*	*New password*

11.3 Video source management

11.3.1 Get source list

URL: /api/v1/GetSrc

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/GetSrc?token=xxx&session=xxxxxxx
Response:
{
 "src": [
  {
            "nType": "H5_ONVIF",
            "strName": "name1",
            "strToken": "token1",
            "strUrl": "rtsp://192.168.0.1/stream",
            "strUser": "admin",
            "strPasswd": "******",
            "bPasswdEncrypt": false,
            "bEnableAudio": false,
            "nConnectType": "H5_AUTO",
            "nRTSPType": "H5_RTSP_AUTO",
            "strSrcIpAddress": "192.168.100.173",
            "strSrcPort": "80",
            "nChannelNumber": 1,
            "bOnvifProfileAuto": true,
            "strOnvifAddr": "/onvif/device_service",
            "strOnvifProfileMain": "Profile_1",
            "strOnvifProfileSub": "Profile_2",
            "bOnline": true,
            "strSnapshotUrl": "/api/v1/GetImage?token=token1",
            "strServerToken": "",//该 token 为对应注册的 h5stream  token
            "strOriginalToken": ""
  }
  ]
 }
}
```

Parameter	Optional/Must	Description
session	*must*	*session id*
token	*optional*	*The source token need to get, is no this parameter, stand for get all.*

11.3.1 Add file source

URL: /api/v1/AddSrcFile

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/
AddSrcFile?name=xxx&token=xxx&url=xxx&session=xxxxxx
Response:
{
 "bStatus": true/false,
 "strCode": "xxxxxx"
}
\r\n
```

Parameter	Optional/Must	Description
session	*must*	*session id*
token	*must*	*source token*
name	*must*	*source name*
url	*must*	*File path in the server, need use url encode to encode* *https://meyerweb.com/eric/tools/dencoder/* *C:\xdev\h5s-r4.0.0403.18-win64-release\h5ssample.mp4* *encode to* *C%3a%5cxdev%5ch5s-r4.0.0403.18-win64-release%5ch5ssample.mp4*
servertoken	*optional*	*Added to h5stream that register to this.(next version)* *And the token will be map to another one, need get again.*

11.3.2 Add RTSP RTMP source

URL: /api/v1/AddSrcRTSP

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/AddSrcRTSP?name=xxx&token=xxx
      &user=admin&password=12345&url=rtsp://192.168.0.1/stream&session=x
xxxxx
Response:
{
 "bStatus": true/false,
 "strCode": "xxxxxx"
}
\r\n
```

Parameter	Optional/Must	Description
session	*must*	*session id*
token	*must*	*source token*
name	*must*	*source name*
user	*optional*	*RTSP username*
password	*must*	*RTSP password*
url	*must*	*File path in the server, need use url encode to encode* *https://meyerweb.com/eric/tools/dencoder/* *rtsp://192.168.0.23:554/Streaming/Channels/101?transportmode=unicast&profile=Profile_1* *Encode to:* *rtsp%3a%2f%2f192.168.0.23%3a554%2fStreaming%2fChannels%2f101%3ftransportmode%3dunicast%26profile%3dProfile_1*
servertoken	*optional*	*Add to register h5stream(next version)*

11.3.3 Add ONVIF source

URL: /api/v1/AddSrcONVIF

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/AddSrcONVIF?name=name1&token=token1
&user=admin&password=12345&ip=192.168.0.1
&port=80&onvifaddr=/onvif/device_service&main=Profile_1&sub=Profile_2&se
ssion=xxxxxx
Response:
{
 "bStatus": true/false,
 "strCode": "xxxxxx"
}
```

Parameter	Optional/Must	Description
session	*must*	*session id*
token	*must*	*source token*
name	*must*	*source name*
user	*must*	*ONVIF username*
password	*must*	*ONVIF password*
ip	*must*	*ONVIF device ip address*
port	*must*	*ONVIF device port*
onvifaddr	*optional*	*ONIVF service address* *Default is* /onvif/device_service
main	*optional*	*Main stream token*
sub	*optional*	*Sub stream token*
servertoken	*optional*	*Add to register h5stream(next version)*

11.3.4 Delete source

URL: /api/v1/DelSrc

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/DelSrc?token=token1&session=xxxxxx
Response:
{
 "bStatus": true/false,
 "strCode": "xxxxxx"
}
\r\n
```

Parameter	Optional/Must	Description
session	*must*	*session id*
token	*must*	*deleted source token*

11.3.5 PTZ

URL: /api/v1/Ptz

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/
Ptz?token=token1&action=left&speed=0.5&session=xxxxxx
Response:
{
 "bStatus": true/false,
 "strCode": "xxxxxx"
}
\r\n
```

Parameter	Optional/Must	Description
session	*must*	*session id*
token	*must*	*ONIVF source token*
action	*must*	*PTZ command* *up/down/left/right/* *zoomin/zoomout/stop* */preset*
speed	*optional*	*stop does not need speed* *speed can be double 0-1*
preset	*optional*	*The preset goto*

11.3.1 Set preset

URL: /api/v1/ SetPreset

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/ SetPreset?token=token1&
&presetname=2&presettoken=2&session=xxxxxx
Response:
{
 "bStatus": true/false,
 "strCode": "xxxxxx"
}
\r\n
```

Parameter	Optional/Must	Description
session	*must*	*session id*
token	*must*	*source token*
presetname	*must*	*Preset name, such as 1，2，3*
presettoken	*must*	*Preset token , such as 1，2，3*

11.3.2 Delete preset

URL: /api/v1/ DelPreset

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/ DelPreset?token=token1& &
presettoken=2&session=xxxxxx
Response:
{
 "bStatus": true/false,
 "strCode": "xxxxxx"
}
\r\n
```

Parameter	Optional/Must	Description
session	*must*	*session id*
token	*must*	*source token*
presettoken	*must*	*preset token, such as 1，2，3*

11.4 Extend ONVIF command

11.4.1 Search ONVIF Device

URL: /api/v1/OnvifSearch

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/OnvifSearch?timeout=10&session=xxxxxx
Response:
{
 "device": [
  {
   "strIp": "192.168.0.23",
   "strPort": "80",
   "strOnvifAddr": "/onvif/device_service",
   "strModel": "xxx-xxx"
  }
 ]
}
```

Parameter	Optional/Must	Description
session	*must*	*session id*
timeout	*optional*	*Searched time，in second* *Default is 10s*
servertoken	*optional*	*Search in the register h5stream(next version)*

11.4.2 Probe ONVIF device

URL: /api/v1/OnvifProbe

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/
OnvifProbe?ip=192.168.0.1&port=80&user=admin&password=12345&onvifaddr=/o
nvif/device_service&session=xxxxxx
Response:
{
 "profile": [
  {
   "strToken": "Profile_1",
   "strCodecName": "H264",
   "strName": "H264 1920 x 1080 25fps 3072bps",
   "strRtspUrl":
"rtsp://192.168.0.23:554/Streaming/Channels/101?transportmode=unicast&pr
ofile=Profile_1",
   "nWidth": 1920,
   "nHeight": 1080,
   "nFps": 25,
   "nBandwidth": 0,
   "bGotUrl": true
  }
 ]
}
Or
{
 "bStatus": false,
 "strCode": "can't connect device"
}
```

Parameter	Optional/Must	Description
session	*must*	*session id*
user	*must*	*ONVIF device username*
password	*must*	*ONVIF device password*
ip	*must*	*ONVIF device ip address*
port	*must*	*ONVIF device port*
onvifaddr	*optional*	*ONVIF service address*
servertoken	*optional*	*Probe in the register h5stream(next version)*

11.5 Record management

11.5.1 Record

URL: /api/v1/Record

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/Record?token=token1&duration=100&filename=
xxxxx&&session=xxxxxx
Response:
{
 "bStatus": true/false,
 "strCode": "xxxxxx"
}
\r\n
```

Parameter	Optional/Must	Description
session	*must*	*session id*
token	*must*	*Source token*
duration	*must*	*Record time*
filename	*optional*	*If the file name is blank, the server auto gen the file name. the file name does not have extend file name. all the record file is mp4.*

11.5.2 Path record

URL: /api/v1/PathRecord

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/PathRecord?token=token1&start1=2018-3-29&start2=3-3-5&tz=8&duration=100&filename=xxxxx&&session=xxxxxx
Response:
{
 "bStatus": true/false,
 "strCode": "xxxxxx"
}
\r\n
```

Parameter	Optional/Must	Description
session	*must*	*session id*
token	*must*	*Source token*
duration	*must*	*Record time*
start1	*must*	*Start time 1* *2018-3-29 y-m-d*
start2	*must*	*Start time 2* *20-3-29 h-m-s* *Client can access the video by blow url* */mediastore/record/token1/2018-3-29TZ8/20-3-29/filename.mp4*
tz	*optional*	*8 or -8, default use the server time zone*
filename	*must*	*Record file name, does not have extend name.*

11.5.3 Snapshot

URL: /api/v1/Snapshot

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/
Snapshot?token=token1&filename=xxxxx&&session=xxxxxx
Response:
{
 "bStatus": true/false,
 "strCode": "xxxxxx"
}
\r\n
```

Parameter	Optional/Must	Description
session	*must*	*session id*
token	*must*	*Source token*
filename	*optional*	*If the file name is blank, the server auto gen the file name. the file name does not have extend file name. all the record file is mp4.*

11.5.4 Path snapshot

URL: /api/v1/PathSnapshot

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/ PathSnapshot?token=token1&start1=2018-3-29&start2=3-3-5&tz=8&filename=xxxxx&&session=xxxxxx
Response:
{
 "bStatus": true/false,
 "strCode": "xxxxxx"
}
\r\n
```

Parameter	Optional/Must	Description
session	*must*	*session id*
token	*must*	*Source token*
start1	*must*	*Start time 1* *2018-3-29 y-m-d*
start2	*must*	*Start time 2* *20-3-29 h-m-s* *Client can access the snapshot by blow url* */mediastore/snapshot/token1/2018-3-29TZ8/20-3-29/filename.jpg*
tz	*optional*	*8 or -8, default use the server time zone*
filename	*must*	*Snapshot file name, does not have extend name.*

11.5.5 Search

URL: /api/v1/Search

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/ Search?type=xxx&start=2018-03-
07T120101%2b08&end=2018-04-26T130101%2b08&token=xxx&session=xxxxxx
Response:
{
    "record": [
        {
            "strToken": "token1",
            "strStartTime": "2018-04-25T19:32:05+08:00",
            "strDuration": "301",
            "strPath": "/mediastore/record/token1/2018-4-25TZ8/19-32-
5/1234.mp4"
        },
        {
            "strToken": "token1",
            "strStartTime": "2018-04-25T19:37:05+08:00",
            "strDuration": "0",
            "strPath": "/mediastore/record/token1/2018-4-25TZ8/19-37-
5/1234.mp4"
        }
    ]
}
\r\n
```

Parameter	Optional/Must	Description
session	*must*	*session id*
token	*must*	*Source token*
type	*must*	*record or snapshot*
start	*must*	*ISO8601 format time* *2018-03-07T120101+08*
end	*must*	*ISO8601 format time* *2018-03-07T130101+08*

11.6 Cloud video management

11.6.1 Get cloud client info

URL: /api/v1/GetCloudClientInfo for internal network H5STREAM register status get.

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/ GetCloudClientInfo&session=xxxxxx
Response:
{
    "bEnable": true,
    "strServerName": "Server 1",
    "strCloudIp": "10.0.0.1",
    "strCloudPort": "8080",
    "bSSL": false,
    "strUser": "admin",
    "bOnline": true
}
\r\n
```

Parameter	Optional/Must	Description
session	*must*	*session id*

11.6.2 Cloud server list get

URL: /api/v1/ GetServerList For cloud H5STREAM to get the server registered to the cloud.

Security level: Administrator, Operator, Viewer

Method: GET

Syntax:

```
Request :http://server/api/v1/GetServerList&session=xxxxxx
Response:
{
    "server": [
        {
            "strServerToken": "fc2f0aa4-86ca-4d75-ad29-59b91326dfae",
            "strServerIp": "192.168.0.1",
            "bOnline": true
        }
    ]
}
\r\n
```

Parameter	Optional/Must	Description
session	*must*	*session id*

12.0 JS API

12.1 JS object Parameter

```
/**
 @param
 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
      hlsver:'v1', //{string} -  v1 is for ts, v2 is for fmp4
      session:'c1782caf-b670-42d8-ba90-2244d0b0ee83' //{string} - session got from login
};
*/
```

12.2 Websocket

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

12.3 WebRTC

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

12.4 HLS

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

12.5 RTMP

For support old version web browser such as IE11, and H5STREAM Flash RTMP player based on videojs, and you can refer www/rtmp.html to a demo.