
Table of Contents

Introduction	1.1
Installing Software	1.2
Configuration	1.3
JavaScript API	1.4
Restful API	1.5
Media Server	1.6
WebRTC	1.7
H5Stream Cluster	1.8
FAQ	1.9

0. h5stream User Manual

.

1. Installing Software

Download & Start

Download the release package from

<https://www.linkingvision.com/download/H5Stream/>

Start the h5ss.bat/h5ss.sh and then use Browser access <http://x.x.x.x:8080> or <https://x.x.x.x:8443> (You can change the port in the configuration)

Windows

Please install vs2017 redistributable

x86

https://download.visualstudio.microsoft.com/download/pr/100349138/88b50ce70017bf10f2d56d60fcb6ab1/VC_redist.x86.exe

x64

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

Run as service

Windows

Run the regservice.bat in the release package

Centos 7

Copy the release package to /opt/h5ss

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

```
systemctl enable h5ss.service
```

```
systemctl start h5ss.service
```

License

You can get your hostid in the logs/h5sslog.log, and send the hostid to info@linkingvision.com, then you can get the license, and copy the h5ss.lic file to conf folder, the h5ss.conf and h5ss.lic both in the conf/ folder.

2. Configuration

Configuration file

In the release package, the `conf/h5ss.conf` is the default configuration file, the configuration is JSON based file, the configuration has two part: System & media source.

You can delete the configuration file, then start the `h5ss.bat/h5ss.sh`, the default configuration file will be generated.

http

Name	Comments
nHTTPPort	HTTP server port, if port is 0, disable it
nHTTPSPort	HTTPS server port, if port is 0, disable it
bAuth	Enable authentication for HTTP/HTTPS

rtsp

Name	Comments
bRTSPSink	Enable RTSP Server
nRTSPPort	RTSP server port
nSSLPort	RTSP over SSL server port
bAuth	Enable authentication for RTSP/RTSP over SSL

rtmp

Name	Comments
bRTMPSink	Enable RTMP Server
nRTMPPort	RTMP server port
nSSLPort	RTMP over SSL server port
bAuth	Enable authentication for RTMP/RTMP over SSL

flv

Name	Comments
bFLVSink	Enable FLV Server
nFLVPort	FLV server port
nSSLPort	FLV over SSL server port
bAuth	Enable authentication for FLV/FLV over SSL

hls

Name	Comments
nHLSSinkType	HLS Sink type H5_HLS_NONE/H5_HLS_V1(ts)/H5_HLS_V2(mp4)
nHLSSegmentNum	HLS Segment number
nHLSDuration	HLS Segment duration

webrtc

Name	Comments
bWebRTCSink	Enable WebRTC sink

system

Name	Comments
nLogType	log type H5_LOG_DEBUG/H5_LOG_INFO/H5_LOG_WARNING/H5_LOG_ERROR
bConsoleLog	Console log enable true/false
nServerThreadNum	Server thread number

user

The user has one or more user, you can refer the comment for the item.

```
"user": {
  "users": [
    {
      "strUserComment": "User name",
      "strUser": "admin",
      "strPasswdComment": "Password",
      "strPasswd": "12345",
      "bPasswdEncryptComment": "Password Encrypted",
      "bPasswdEncrypt": false,
      "strUserTypeComment": "User type Administrator/Standard",
      "strUserType": "Administrator"
    }
  ]
}
```

source

The src has one or more source, you can refer the comment for the item.

```
"source": {
  "bEnableAudioComment": "Enable Audio",
  "bEnableAudio": false,
  "nConnectTypeComment": "H5_ONDEMAND/H5_ALWAYS",
  "nConnectType": "H5_ALWAYS",
  "nRTSPTypeComment": "RTSP Connect protocol H5_RTSP_TCP/H5_RTSP_UDP/H5_RTSP_HTTP/H5_RTSP_HTTPS",
  "nRTSPType": "H5_RTSP_TCP",
  "src": [
    {
      "strNameComment": "name for this stream",
      "strName": "Stream 1",
      "strTokenComment": "token for this stream, must unique, if same, only frist will be available",
      "strToken": "token1",
      "nTypeComment": "source type H5_FILE/H5_STREAM",
      "nType": "H5_STREAM",
      "strUrlComment": "",
      "strUrl": "rtsp://192.168.0.1/stream",
      "strUserComment": "username",

```

```
"strUser": "admin",
"strPasswdComment": "password",
"strPasswd": "12345",
"bPasswdEncryptComment": "Password Encrypted",
"bPasswdEncrypt": false
},
{
"strNameComment": "name for this stream",
"strName": "Stream 2",
"strTokenComment": "token for this stream, must unique, if s
ame, only frist will be available",
"strToken": "token2",
"nTypeComment": "source type H5_FILE/H5_STREAM",
"nType": "H5_STREAM",
"strUrlComment": "",
"strUrl": "rtsp://192.168.0.1/stream",
"strUserComment": "username",
"strUser": "admin",
"strPasswdComment": "password",
"strPasswd": "12345",
"bPasswdEncryptComment": "Password Encrypted",
"bPasswdEncrypt": false
}
]
}
```


3. JavaScript API

H5sPlayerWS

```
/**
 * Connect a websocket Stream to videoElement
 */
H5sPlayerWS.prototype.connect

/**
 * Disconnect a websocket Stream and clear videoElement source
 */
H5sPlayerWS.prototype.disconnect
```

H5sPlayerHLS

```
/**
 * Connect a HLS Stream to videoElement
 */
H5sPlayerHLS.prototype.connect

/**
 * Disconnect a HLS Stream and clear videoElement source
 */
H5sPlayerHLS.prototype.disconnect
```

Example

```
<!DOCTYPE HTML>
<html>
<head>
<title>HTML5 Video</title>
<script src="/js/jquery-3.1.1.js"></script>
<script src="/js/bootstrap.js"></script>
<script src="/js/h5splayer.js"></script>
<link rel = "stylesheet" type = "text/css" href = "/css/h5splaye
```

```
r.css" />

</head>

<body>

<div class="wrapper">
  <video class="video" id="h5sVideo1" width="640" height="360"
  " autoplay style="background-color:#000000">

  </video>
  <div class="playpause"></div>
</div>

<div class="wrapper">
  <video class="video" id="h5sVideo2" width="640" height="360"
  autoplay style="background-color:#000000">

  </video>
  <div class="playpause"></div>
</div>
</body>

<script>
var v1;
var v2;

if (H5siOS())
{
  v1 = new H5sPlayerHls("h5sVideo1", "token1", "v1");
  v2 = new H5sPlayerHls("h5sVideo2", "token2", "v1");
}else
{
  v1 = new H5sPlayerWS("h5sVideo1", "token1");
  v2 = new H5sPlayerWS("h5sVideo2", "token2");
}

$('#h5sVideo1').parent().click(function () {
  if($(this).children(".video").get(0).paused){
    console.log(v1);
```

```
        v1.connect("token1");
        $(this).children(".video").get(0).play();
        $(this).children(".playpause").fadeOut();
    }else{
        v1.disconnect();
        $(this).children(".video").get(0).pause();
        $(this).children(".playpause").fadeIn();
    }
});

$('#h5sVideo2').parent().click(function () {
    if($(this).children(".video").get(0).paused){
        console.log(v2);
        v2.connect("token2");
        $(this).children(".video").get(0).play();
        $(this).children(".playpause").fadeOut();
    }else{
        v2.disconnect();
        $(this).children(".video").get(0).pause();
        $(this).children(".playpause").fadeIn();
    }
});

</script>

</html>
```

4. Restful API

Add Stream

<http://x.x.x.x:8080/hapi/AddStream>

Method:POST

Request:

```
{
  "strName": "Stream 1",
  "strToken": "tokenabc",
  "strUrl": "rtsp://192.168.0.1/stream",
  "strUser": "admin",
  "strPasswd": "12345"
}
```

Response:

```
{
  "bStatus": true,
  "strCode": "Add successfully"
}
```

Delete Stream

<http://x.x.x.x:8080/hapi/DelStream>

Method:POST

Request:

```
{
  "strToken": "tokenabc"
}
```

Response:

```
{
  "bStatus": true,
  "strCode": "Delete successfully"
}
```


5. Media Server

H5Stream has build in RTSP RTMP FLV server, the format url is like below, you can change the port in the h5ss.conf file.

`rtsp://x.x.x.x:8554/live/token1`

`rtmp://x.x.x.x:8935/live/token1`

<http://x.x.x.x:8090/live/token1.flv>

5. WebRTC

5.H5Stream Cluster

How to enable HLS ?

Open the h5ss.conf file, change the nConnectType from H5_ONDEMAND to H5_ALWAYS

and then change the nHLSSinkType to H5_HLS_V1

- These has video display ?

Check the password and user in the source.

- How to use curl to add source ?

Below is the windows bat file format, you change to linux or macOS

```
set HAPI_HOST=192.168.1.2:8080
set HAPI_URL_ADD=http://%HAPI_HOST%/hapi/AddStream
set HAPI_URL_DEL=http://%HAPI_HOST%/hapi/DelStream
set STREAM_URL="rtsp://192.168.1.1:8554/live/token1\"

curl -H "Content-Type: application/json" -X POST -d '{ "strName": "Stream 1", "strToken": "token100", "strUrl": %STREAM_URL%, "strUser": "admin", "strPasswd": "12345"}' %HAPI_URL_ADD%
curl -H "Content-Type: application/json" -X POST -d '{"strToken": "token1"}' %HAPI_URL_DEL%
```