IP Camera audio & video protocol

This documents describe how to get videostream and other media data from IP cameras . IP Cameras support both public and private communication protocols

1. Public communication protocol RTSP Over Http

Quicktime support this protocol . Pls find more details in RTSP ,RTP and HTTP protocol.

2. Public communication protocol RTSP Over UDP

Quicktime ,.VLC support this protocol . Pls find more details in RTSP and RTP protocol.

3. Private communication protocol

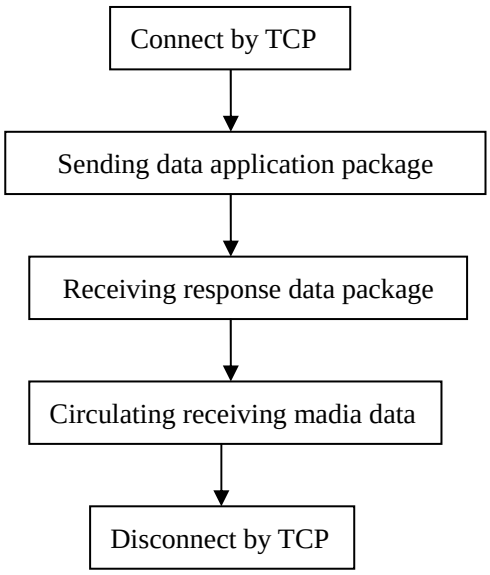
Get video stream by using this protocol

Media data include video data , audio data ,motion detection data, external alarm data.

This protocol is based on TCP/IP.

3.1 Flow Chart of Communication Protocol

1). Flow Chart for receiving and sending media data



3.2 Detailed Description

1). Media Data

a.Application Package for Media Data

GET http://[IP]:[port]/livestream/[number]?action=play&media=[type] HTTP/1.1\r\n
User-Agent: HiIpcam/V100R003 VodClient/1.0.0\r\n
Connection: Keep-Alive\r\n
Cache-Control: no-cache\r\n
Authorization: [username] [password] \r\n
Content-Length: [length] \r\n
\r\n
Cseq: 1\r\n
Transport: RTP/AVP/TCP;unicast;interleaved=0-1\r\n
\r\n

Fields	Explanation
IP	IP Address
port	Port
number	Bit stream number 11: First channel chief bit stream 12: First channel minor bit stream 21: Second channel chief bit stream 22: Second channel minor bit stream NOTE: Existing equipment has only first channel bit stream
type	request data type video only Video data audio only Audio data data Alarm data (don't support now) video_audio Video and Audio data video_data Video and Alarm data audio_data Audio and Alarm data video_audio_data Audio , Video and Alarm data
username	Username
password	Password
length	Data length after the field. (number of bytes from Cseq), support strlen() function.

Example:
GET http://192.168.1.88:80/livestream/11?action=play&media=video_audio_data HTTP/1.1\r\n
User-Agent: HiIpcam/V100R003 VodClient/1.0.0\r\n
Connection: Keep-Alive\r\n
Cache-Control: no-cache\r\n
Authorization: guest guest\r\n
Content-Length: 57\r\n
Cseq: 1\r\n
Transport: RTP/AVP/TCP;unicast;interleaved=0-1\r\n

\r\n\

b. Media data response package

1). Checkout Username and password

Example:

HTTP/1.1 401 Unauthorized\r\n

Host: 192.168.1.88\r\n

Connection: Keep-Alive\r\n

2). Authority confirmed , Return to Video and Audio types

HTTP/1.1 200 OK\r\n

Host: [IP]\r\n

Connection: Keep-Alive\r\n

Server: HiIpcam/V100R003 VodServer/1.0.0\r\n

Cache-Control: no-cache\r\n

Accept-Ranges: Bytes\r\n

Content-Type: application/octet-stream\r\n

Connection: close\r\n

\r\n

Session: 15547656\r\n

Cseq: 1\r\n

m=video 96 H264/90000/[Width]/[Height]\r\n

m=audio 97 G726/8000/1\r\n

Transport: RTP/AVP/TCP;unicast;hisiinterleaved=0-1;ssrc=614fd4a1\r\n

\r\n

Fields	Explanation
video	Width： Width of the Video data Height： Height of the Video data Only H.264 Video data
audio	97 G726 8 G711a All audio data is single track, 8k sample ,16 bit wide G711a bit stream 64kbps G726 bit stream 16kbps

Example:

HTTP/1.1 200 OK\r\n

Host: 192.168.1.88\r\n

Connection: Keep-Alive\r\n

Server: HiIpcam/V100R003 VodServer/1.0.0\r\n

Cache-Control: no-cache\r\n

Accept-Ranges: Bytes\r\n

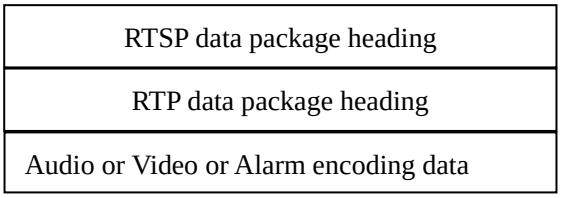
Content-Type: application/octet-stream\r\n

Connection: close\r\n

\r\n
Session: 15547656\r\n
Cseq: 1\r\n
m=video 96 H264/90000/704/576\r\n
m=audio 97 G726/8000/1\r\n
Transport: RTP/AVP/TCP;unicast;hisiinterleaved=0-1;ssrc=614fd4a1\r\n
\r\n
NOTE: Make sure that judge the end of HTTP package by following HTTP protocol.

c. Content of Media data package

1). Structure chart of Data package heading



1. Format of RTSP data package

```
typedef struct
{
    Unsigned char   daollar;        /*8, $:dollar sign(24 decimal)*/
    Unsigned char   channelid;      /*8, channel id*/
    Unsigned short resv;            /*16, reseved*/
    Unsigned int payloadLen;    /*32, payload length*/
    RTP_HDR_S rtpHead;    /*rtp head*/
}RTSP_ITLEAVED_HDR_S;
```

Fields	Explanation
payloadLen	Length of RTP data (Include the length of RTP_HDR_S) NOTE: This filed is Net-Endianness
Cannot deal with other value	

2. Format of RTP data package heading.

```
typedef struct
{
    /* byte 0 */
    Unsigned short cc          :4;    /* CSRC count */
    Unsigned short x           :1;    /* header extension flag */
    Unsigned short p           :1;    /* padding flag */
    Unsigned short version :2;    /* protocol version */
    /* byte 1 */
    Unsigned short pt          :7;    /* payload type */
    Unsigned short marker   :1;    /* marker bit */
```

```
    /* bytes 2, 3 */
    Unsigned short seqno    :16;     /* sequence number */
    /* bytes 4-7 */
    Unsigned int ts;                 /* timestamp in ms */
    /* bytes 8-11 */
    Unsigned int ssrc;               /* synchronization source */
} RTP_HDR_S;
```

Fields	Explanation
pt	96 H.264 97 G.726 8 G.711a 100 Alarm data
ts	Time stamp (msec) NOTE: This filed is Net-Endianness
Cannot deal with other value	

3. Video encoding data
   The video data will be a Nalu package of H.264 when received, It may be a full frame, also may be not. If several Nalu package compose a full frame, it will need framing operation.
   There is a simple method to framing, according the timestamp, if have same timestamp, they can be a frame.

4. Audio encoding data
   Make sure to cut off 4 bytes of private data heading when encode video data
5. Alarm data
   Motion Detection Content
   When it is alarming in all four alarm areas:

"MDName1-x-y-width-height| MDName2-x-y-width-height| MDName3-x-y-width-height|MDName4-x-y-width-height|"

Fields	Explanation
MDName	移动检测区域名称(值：1, 2, 3, 4) Motion Detection Area name(value: 1,2,3,4)
x	X ordinate
y	Y ordinate
width	width
height	height
"|"	data break note

Example:
1-10-10-20-20|

External Alarm data content.
alarmin:1