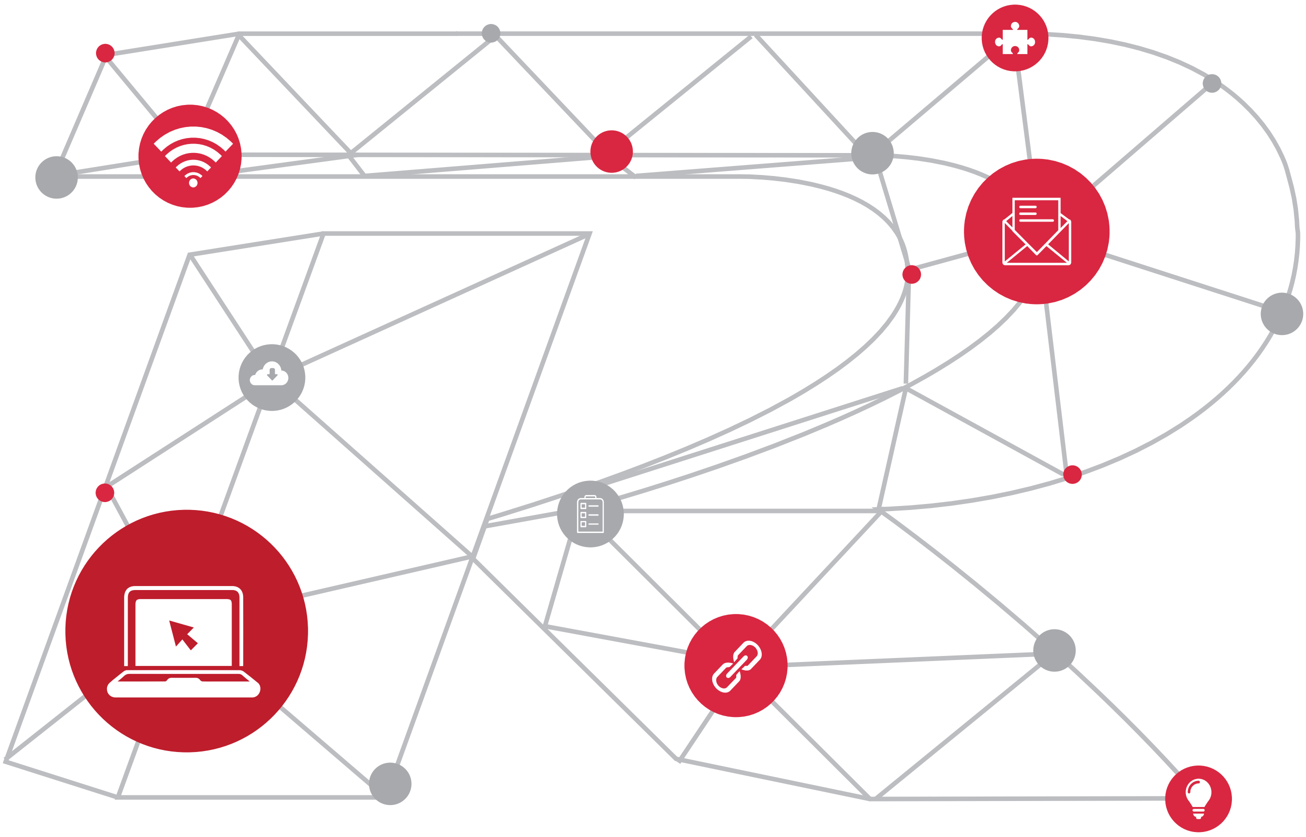
Ruijie IP Surveillance

White Paper



Contents

Contents

Introduction

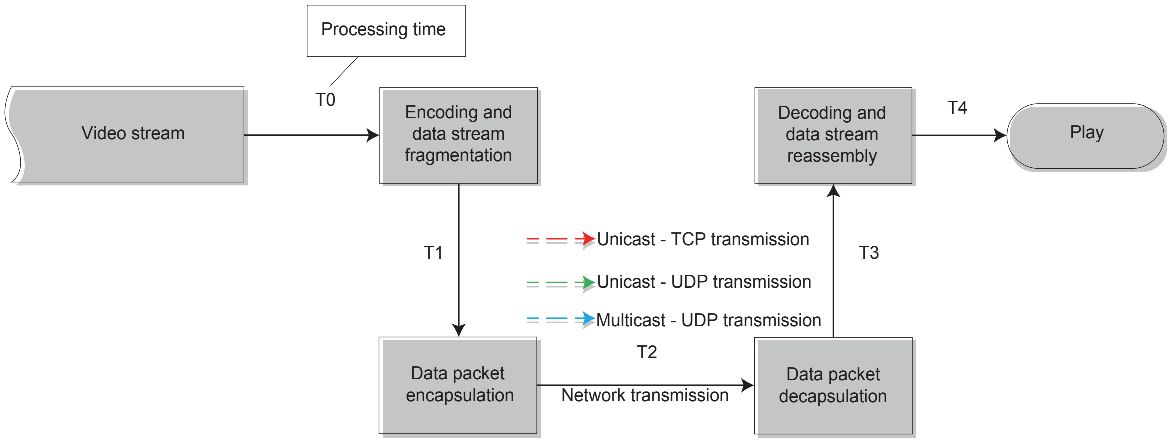
This document describes the functions of network devices in an IP surveillance system in terms of transmission. After analyzing the strong burst feature of video data streams that are output by IP Cameras (IPCs), this document provides some targeted optimization means. Ruijie switches are capable of optimizing video data transmission and can properly alleviate and even eliminate stuttering, pixelation, and other common problems occurring in the current IP surveillance industry.

An IP surveillance system generally includes three sections: front end, transmission, and rear end. Data generated by the front end is transmitted to the rear end via the transmission network, and the rear end stores, processes, and analyzes the data. However, the function requirements of each section change with the service development and sector differences. For example, in unattended ATM scenarios in the finance sector, front-end equipment often needs to perform intelligent video analysis and identify events requiring special attention. In addition, an IP surveillance system is composed of products from a surveillance product supplier and products from a network device supplier. This document describes below the functions of network devices in an IP surveillance system in terms of transmission.

The transmission section mainly transfers video data collected by the front end to the rear end rapidly in integrity. Some specifications pose requirements for transmission devices, such as packet loss rate and delay. To better understand these requirements, it is necessary to learn about how video data collected by the front end is transmitted and learn about the video data itself.

The figure below shows the processing of video data.

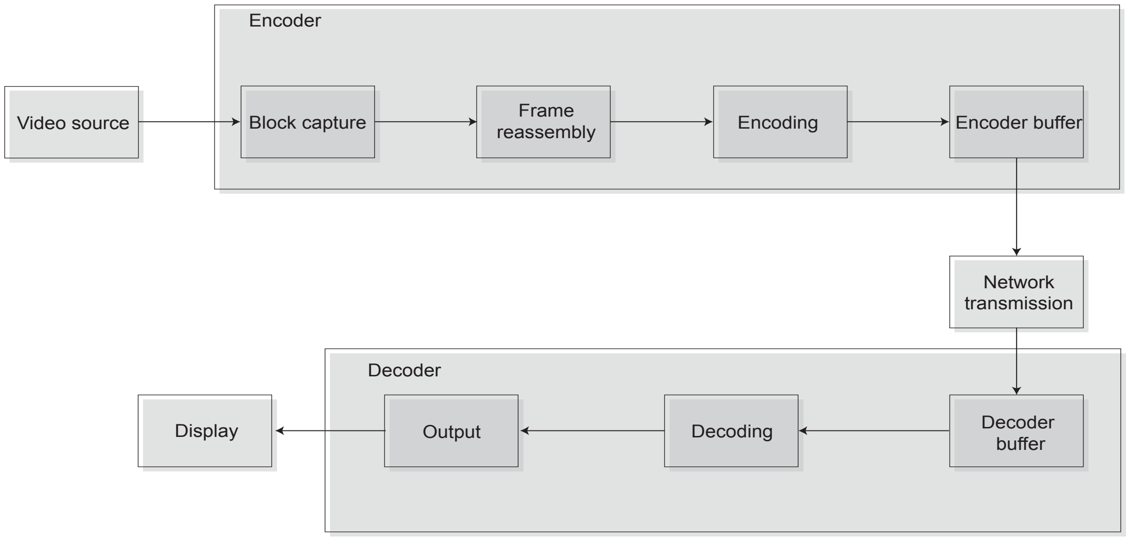
Figure 1


Processing time
T0
Video stream
Encoding and data stream fragmentation
Decoding and data stream reassembly
T4
Play
T1
Unicast - TCP transmission
Unicast - UDP transmission
Multicast - UDP transmission
T3
T2
Data packet encapsulation
Network transmission
Data packet decapsulation

The end-to-end IP surveillance process includes collection, encoding, transmission, decoding, and play. Video data is encoded using the H.264 protocol (mainstream). Transmission Control Protocol (TCP) unicast, User Datagram Protocol (UDP) unicast, and UDP multicast are used for transmission. Real-Time Transport Protocol (RTP) is adopted at the application layer to ensure the real-time performance of data.

The figure below shows the conventional end-to-end IP surveillance process.

Figure 2



Concepts

• Video Data

A video stream that seems consecutive to human eyes is actually composed of static frames. When viewing visual images generated by optical signals, human eyes experience the visual staying phenomenon for about 1/24 seconds. Therefore, frames of video data can be played continuously one after another to deliver a continuous visual effect.

Picture quality is an important attribute of video data. Pictures are composed of pixels, and more pixels can present a more vivid and rich picture. A color is generally represented with color level. A more complex color level system of pixels indicates richer colors. The data amount is calculated using the following formula: Data amount = Pixel quantity x Size of color information of each pixel.

Use an IP network as an example. The foregoing analysis shows that data is generated at an interval of 1/24 seconds during the transmission of video data, and more pixels lead to more complex color levels and more to-be-generated data. If the generated data cannot be thoroughly transmitted within 1/24 seconds, not all of the data cannot be transmitted to the receive end. In this case, some pixel information or some color level information will be lost, and pixilation or frame dropping may occur. Therefore, it is originally desired that video data can be transmitted via Internet to reduce the transmission pressure. A video encoding and compression algorithm is formulated to compress data to

• Data Processing on IPCs

An IPC serves as a video data collector, which captures pictures via photosensitive elements such as the Charge Coupled Device (CCD) and Complementary Metal-Oxide-Semiconductor (CMOS), conducts digital-to-analog conversion on the pictures, and compresses the pictures by using a video encoding and compression algorithm, to generate data suitable for transmission via the transmission network.

Sampling needs to be performed at a rate of 24 frames per second (or other preset frame rates) to ensure continuous pictures. However, the IP surveillance system may encounter exceptions, for example, an interruption in the system may prolong the data processing of one frame. Besides, poor processing by an encoding and compression algorithm on a certain type of pictures may cause a size of generated data to be far greater than the nominal value.

Video data streams received by the transmission network are not always smooth and stable. An encoding buffer is usually set for IPCs to reduce such exceptions. The bit rate may become unstable after video encoding. The transmission of some frames through fixed channels possibly cannot be completed within the defined time, while some frames have spare transmission time. Consequently, the decoder often fails to decode all frames within the defined time, resulting in unsmooth pictures and unstable bit rate. The bitstream of an encoded video is often unstable. After bitstream data is buffered in the encoding buffer and the bitstream is equalized prior to transmission, the bitstream stability is acceptable to the decoder. The encoding buffer cannot be full during video decoding. If it is almost full but data in the encoding buffer cannot be sent out while the encoder is still encoding data, the encoding buffer will become full soon and buffer overflow may occur.

• H.264 and RTP

Video data is encoded using the H.264 protocol and transmitted via the RTP. The following are several terms relevant to H.264 and RTP.

* **I frame: a complete key frame, which retains a complete picture and does not reference other frames to decode.**

* **P frame: contains only data different from the previous frame and references the previous I frame or P frame to decode. A P frame, also known as the difference frame, indicates the difference between it and the previous key frame (or P frame). The previously buffered picture plus the difference defined by this frame is required to generate a final picture during decoding. The P frame, also known as the difference frame, contains only the data different from the previous frame.**

* **GOP length: number of frames between two I frames.**

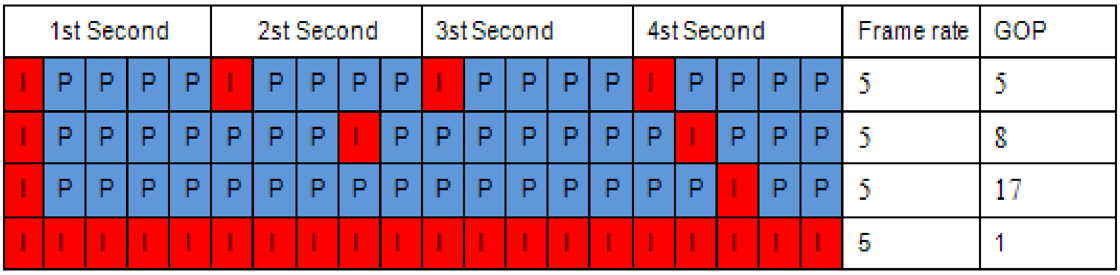
* **Frame rate: frames per second.**

* **Bit rate: sampling frequency, that is, amount of data generated per second.**

The decoding of I frames and P frames is simple and occupies relatively few resources. During I frame decoding, the decoder only needs to decode the current I frame. During P frame decoding, the decoder only needs to decode the current P frame together with one previously buffered frame. If a video stream contains only I frames and P frames, the decoder can concurrently read and linearly decode frames regardless of subsequent data.

By default, both the frame rate and GOP length are the same for IPCs. The GOP length can be increased to reduce the number of I frames in the same period of time. In this way, the bandwidth consumption and storage space consumption can be reduced because I frames are much larger than P frames. Generally, the compression ratio is 7 for I frames (similar to JPG pictures) and 20 for P frames. The compression ratio is relevant to the bit rate and frame rate. When the bit rate is constant, a higher frame rate indicates a larger compression ratio.

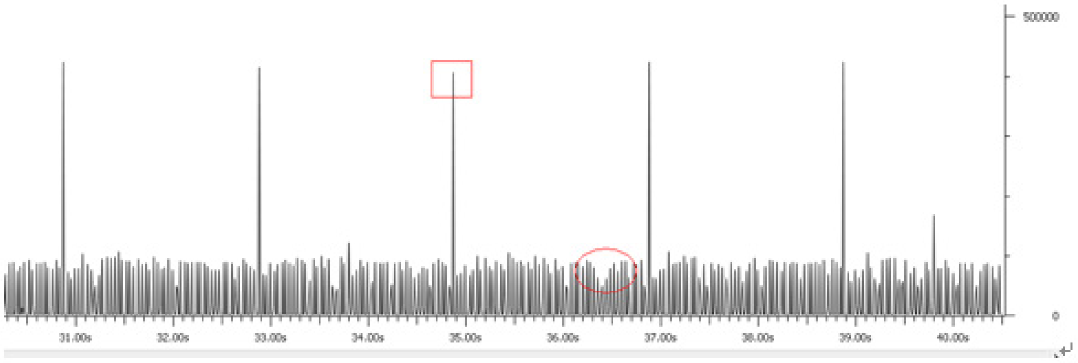
Figure 3



1st Second					2st Second					3st Second					4st Second					Frame rate	GOP
I	P	P	P	P	I	P	P	P	P	I	P	P	P	P	I	P	P	P	P	5	5
I	P	P	P	P	P	P	P	I	P	P	P	P	P	P	P	I	P	P	P	5	8
I	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	I	P	P	5	17
I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	5	1

The figure below shows the traffic of data transmitted by Dahua HF5200 IPC via TCP at the bit rate of 2 Mbps, frame rate of 25 fps, and resolution of 720P. Strong bursts (marked in a red rectangle) occur in I frames while other burrs (marked in a red circle) are P frames.

Figure 4

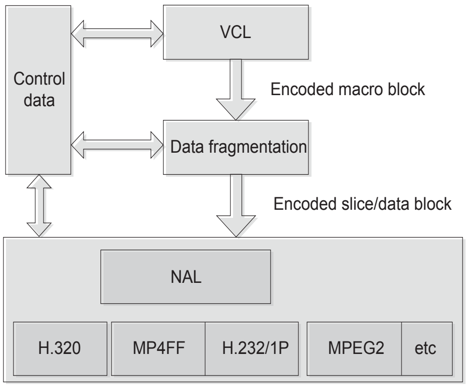


The preceding figure shows that I frames experience much stronger traffic burst than P frames. The figure below shows the H.264 video data processing specified in RFC 3984.

1. H.264 Video Coding Layer (VCL): implements picture compression, frame prediction, and other functions.

2. H.264 Network Abstraction Layer (NAL): uses the segmentation format of the lower layer network to encapsulate data, including the signaling for multiplexed frames and logical channels or transmission of sequence end signals. The NAL transmits encoded data in NAL Units (NALUs) over a packet switched network, and NALUs are conducive to the transmission of packetized data in the network. For bitstream-oriented and packet-oriented transmission, NALUs adopt a unified data format and each NALU contains a single-byte packet header and multi-byte data. The packet header carries the storage flag and type flag. The storage flag indicates whether current data belongs to the referenced frame so that the server discards the data based on the network congestion status. The type flag indicates the type of the image data.

Figure 5



Compression method adopted by H.264:

* **Grouping: distributes several frames of pictures to one GOP, that is, one sequence. The frames in a GOP cannot be excessive to prevent motion changes.**

* **Frame definition: defines various frames in a GOP into three types: I frames, B frames, and P frames.**

* **Frame prediction: predicts P frames based on I frames, and predicts B frames based on I frames and P frames.**

* **Data transmission: stores and transmits I frames and predicted difference information.**

Intra-frame compression is also called spatial compression. When a frame is compressed, only the data of the current frame is considered, and redundant information between adjacent frames is not considered, which is similar to static picture compression. Intra-frame compression often uses the lossy compression algorithm. A complete picture is encoded in intra-frame compression, and therefore, the picture can be decoded and displayed independently. The intra-frame compression ratio cannot be very high and is similar to that of JPEG picture encoding.

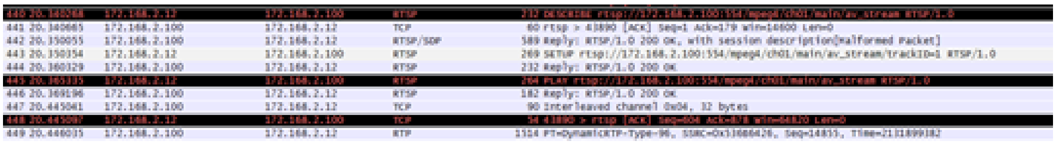
The theory of inter-frame compression is as follows: The data of several adjacent frames is highly related or the information difference between two consecutive frames is small. That is, redundant information exist between adjacent video frames. The redundant information between adjacent frames can be compressed to further increase the compression amount and reduce the compression ratio. Inter-frame compression is also referred to as temporal compression, which compresses frames by comparing data of different frames on the time axis. Inter-frame compression is generally lossless. Frame differencing algorithm is a typical temporal compression method. It compares the current frame and adjacent frames and records only the differences between them, which greatly reduces the data amount.

Technical Principle

• Transmission Process

IP surveillance adopts the Real-Time Streaming Protocol (RTSP)/RTP protocol for data transmission and play. When the main console client establishes a connection to an IPC, it conducts negotiation and transmits data via RTSP.

Figure 6



```
440 20.340268  172.168.2.12     172.168.2.100   RTSP       212 DESCRIBE rtsp://172.168.2.100:554/mpeg4/ch01/main/av_stream RTSP/1.0
441 20.340665  172.168.2.100    172.168.2.12    TCP         60 rtsp > 43890 [ACK] Seq=1 Ack=179 Win=14600 Len=0
442 20.350055  172.168.2.100    172.168.2.12    RTSP/SDP   589 Reply: RTSP/1.0 200 OK, with session description[Malformed Packet]
443 20.350354  172.168.2.12     172.168.2.100   RTSP       269 SETUP rtsp://172.168.2.100:554/mpeg4/ch01/main/av_stream/trackID=1 RTSP/1.0
444 20.360329  172.168.2.100    172.168.2.12    RTSP       232 Reply: RTSP/1.0 200 OK
445 20.365335  172.168.2.12     172.168.2.100   RTSP       264 PLAY rtsp://172.168.2.100:554/mpeg4/ch01/main/av_stream RTSP/1.0
446 20.369196  172.168.2.100    172.168.2.12    RTSP       182 Reply: RTSP/1.0 200 OK
447 20.445041  172.168.2.100    172.168.2.12    TCP         90 Interleaved channel 0x04, 32 bytes
448 20.445097  172.168.2.12     172.168.2.100   TCP         54 43890 > rtsp [ACK] Seq=604 Ack=878 Win=64820 Len=0
449 20.446035  172.168.2.100    172.168.2.12    RTP       1314 PT=DynamicRTP-Type-96, SSRC=0x31666426, Seq=14833, Time=2131899182
```

RTSP is generally based on TCP for transmission and uses a structure similar to the Hypertext Transfer Protocol (HTTP).

- **Request first-line**: add DESCRIBE, SETUP.

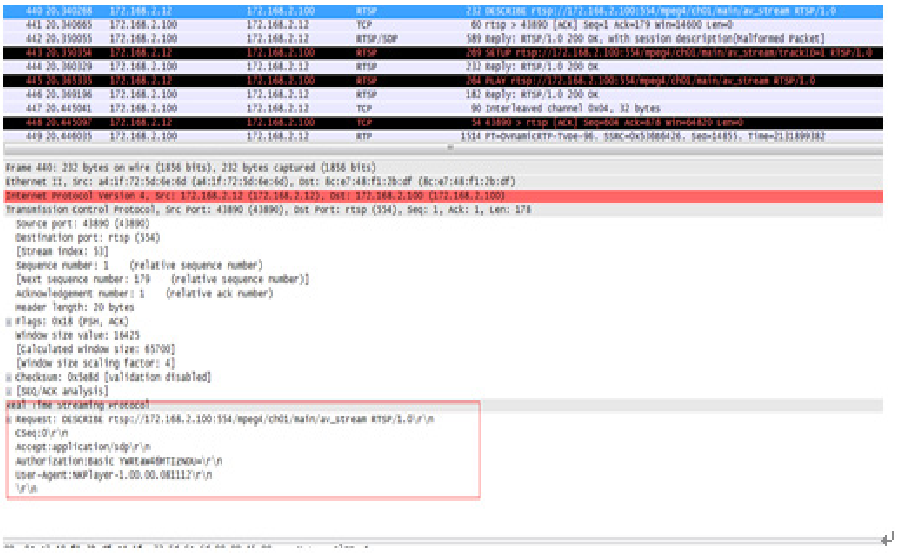
- **Reply first-line**: for example, RTSP/1.0 200 OK.

- **Header field**: in the format of "key : value \r\n" and ended with "\r\n".

- **Body content**: The format is unlimited and the size is determined by **Content-Length**.

The IP surveillance system first calls the **DESCRIBE** command of RTSP to obtain the media description information during IP surveillance.

Figure 7

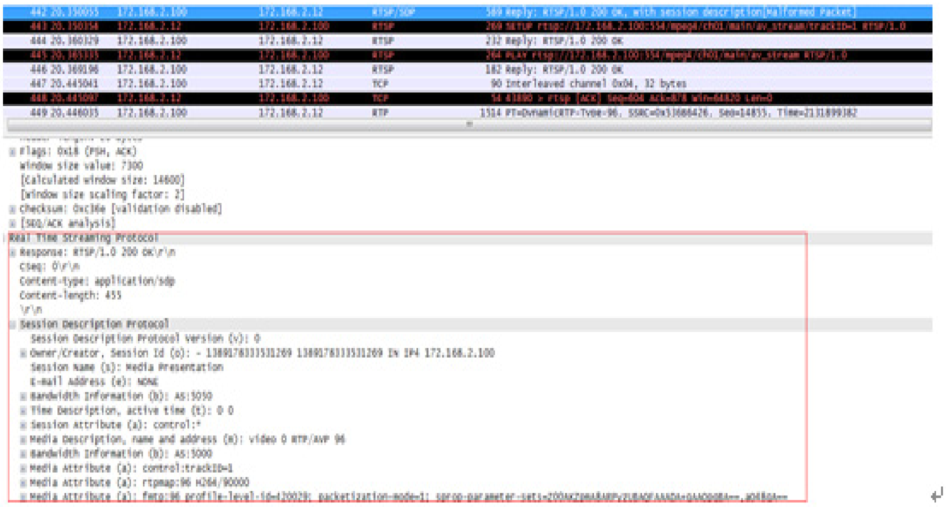


```
440 20.340268   172.168.2.12    172.168.2.100   RTSP      232 DESCRIBE rtsp://172.168.2.100:554/mpeg4/ch01/main/av_stream RTSP/1.0
441 20.340685   172.168.2.100   172.168.2.12    TCP        60 rtsp > 43890 [ACK] Seq=1 Ack=179 Win=14600 Len=0
442 20.350055   172.168.2.100   172.168.2.12    RTSP/SDP  589 Reply: RTSP/1.0 200 OK, with session description[Malformed Packet]
443 20.350354   172.168.2.12    172.168.2.100   RTSP      269 SETUP rtsp://172.168.2.100:554/mpeg4/ch01/main/av_stream/trackID=1 RTSP/1.0
444 20.360329   172.168.2.100   172.168.2.12    RTSP      232 Reply: RTSP/1.0 200 OK
445 20.365335   172.168.2.12    172.168.2.100   RTSP      264 PLAY rtsp://172.168.2.100:554/mpeg4/ch01/main/av_stream RTSP/1.0
446 20.369196   172.168.2.100   172.168.2.12    RTSP      182 Reply: RTSP/1.0 200 OK
447 20.445041   172.168.2.100   172.168.2.12    TCP        90 Interleaved channel 0x04, 32 bytes
448 20.445097   172.168.2.12    172.168.2.100   TCP        54 43890 > rtsp [ACK] Seq=604 Ack=878 Win=64820 Len=0
449 20.446035   172.168.2.100   172.168.2.12    RTP      1514 PT=DynamicRTP-Type-96, SSRC=0x53886426, Seq=14855, Time=2131899382

Frame 440: 232 bytes on wire (1856 bits), 232 bytes captured (1856 bits)
Ethernet II, Src: a4:1f:72:5d:6e:6d (a4:1f:72:5d:6e:6d), Dst: 8c:e7:48:f1:2b:df (8c:e7:48:f1:2b:df)
Internet Protocol Version 4, Src: 172.168.2.12 (172.168.2.12), Dst: 172.168.2.100 (172.168.2.100)
Transmission Control Protocol, Src Port: 43890 (43890), Dst Port: rtsp (554), Seq: 1, Ack: 1, Len: 178
  Source port: 43890 (43890)
  Destination port: rtsp (554)
  [Stream index: 53]
  Sequence number: 1    (relative sequence number)
  [Next sequence number: 179    (relative sequence number)]
  Acknowledgement number: 1    (relative ack number)
  Header length: 20 bytes
⊞ Flags: 0x18 (PSH, ACK)
  Window size value: 16425
  [Calculated window size: 65700]
  [Window size scaling factor: 4]
⊞ Checksum: 0x5e8d [validation disabled]
⊞ [SEQ/ACK analysis]
Real Time Streaming Protocol
⊞ Request: DESCRIBE rtsp://172.168.2.100:554/mpeg4/ch01/main/av_stream RTSP/1.0\r\n
  CSeq:0\r\n
  Accept:application/sdp\r\n
  Authorization:Basic YWRtaW46MTIzNDU=\r\n
  User-Agent:NKPlayer-1.00.00.081112\r\n
  \r\n
```

After receiving the message, the IPC responds accordingly.

Figure 8



```
442 20.350055   172.168.2.100   172.168.2.12    RTSP/SDP  589 Reply: RTSP/1.0 200 OK, with session description[Malformed Packet]
443 20.350354   172.168.2.12    172.168.2.100   RTSP      269 SETUP rtsp://172.168.2.100:554/mpeg4/ch01/main/av_stream/trackID=1 RTSP/1.0
444 20.360329   172.168.2.100   172.168.2.12    RTSP      232 Reply: RTSP/1.0 200 OK
445 20.365335   172.168.2.12    172.168.2.100   RTSP      264 PLAY rtsp://172.168.2.100:554/mpeg4/ch01/main/av_stream RTSP/1.0
446 20.369196   172.168.2.100   172.168.2.12    RTSP      182 Reply: RTSP/1.0 200 OK
447 20.445041   172.168.2.100   172.168.2.12    TCP        90 Interleaved channel 0x04, 32 bytes
448 20.445097   172.168.2.12    172.168.2.100   TCP        54 43890 > rtsp [ACK] Seq=604 Ack=878 Win=64820 Len=0
449 20.446035   172.168.2.100   172.168.2.12    RTP      1514 PT=DynamicRTP-Type-96, SSRC=0x53886426, Seq=14855, Time=2131899382

  Header length: 20 bytes
⊞ Flags: 0x18 (PSH, ACK)
  Window size value: 7300
  [Calculated window size: 14600]
  [Window size scaling factor: 2]
⊞ Checksum: 0xc36e [validation disabled]
⊞ [SEQ/ACK analysis]
Real Time Streaming Protocol
⊞ Response: RTSP/1.0 200 OK\r\n
  CSeq: 0\r\n
  Content-type: application/sdp
  Content-length: 455
  \r\n
⊟ Session Description Protocol
    Session Description Protocol Version (v): 0
  ⊞ Owner/Creator, Session Id (o): - 1389178333531269 1389178333531269 IN IP4 172.168.2.100
    Session Name (s): Media Presentation
    E-mail Address (e): NONE
  ⊞ Bandwidth Information (b): AS:5050
  ⊞ Time Description, active time (t): 0 0
  ⊞ Session Attribute (a): control:*
  ⊞ Media Description, name and address (m): video 0 RTP/AVP 96
  ⊞ Bandwidth Information (b): AS:5000
  ⊞ Media Attribute (a): control:trackID=1
  ⊞ Media Attribute (a): rtpmap:96 H264/90000
  ⊞ Media Attribute (a): fmtp:96 profile-level-id=420029; packetization-mode=1; sprop-parameter-sets=Z00AKZpmA8ARPyZUBAQFAAADA+gAAOpgBA==,aO48gA==
```

The Session Description Protocol (SDP) is a session description format rather than a transmission protocol. SDP does not support the negotiation of session content or media code, and is used only to describe media information in media streaming.

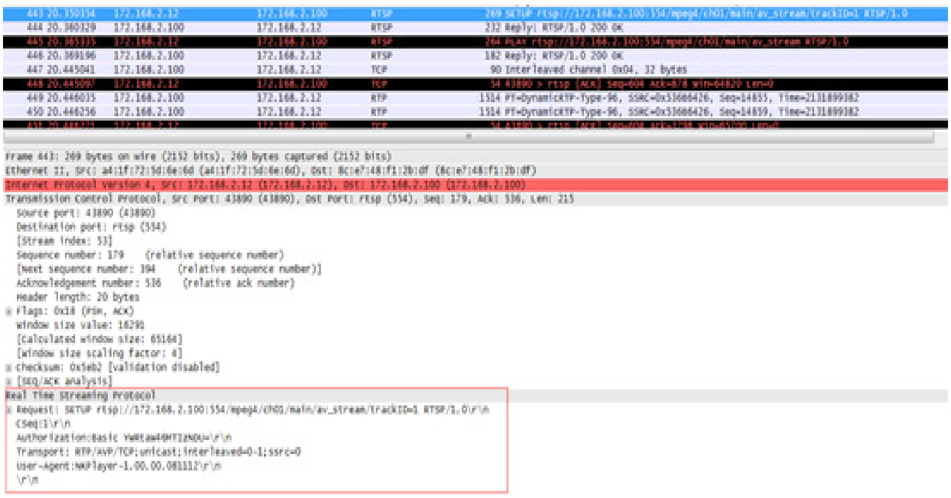
rtpmap: RTP packet description, encapsulated in the H264 format. The value is 96 H264/90000, where 90000 indicates the clock frequency. The timestamp of each packet is calculated based on this clock frequency. One frame may be transferred via multiple packets and the multiple packets share the same timestamp. The timestamps of frames are different and increase by 3600 (90000/25) in the case of 25 frames.

control: control information of tracks (channel).

fmtp: media format-relevant parameters, for transmitting encapsulation- and decoding-relevant information.

The second step is the setup process.

Figure 9



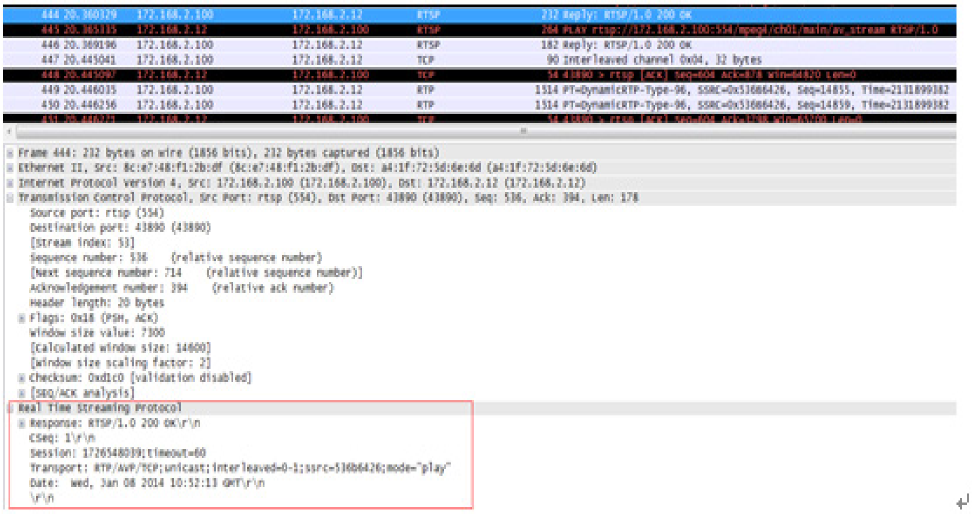
```
443 20.350354  172.168.2.12   172.168.2.100  RTSP  269 SETUP rtsp://172.168.2.100:554/mpeg4/ch01/main/av_stream/trackID=1 RTSP/1.0
444 20.360329  172.168.2.100  172.168.2.12   RTSP  232 Reply: RTSP/1.0 200 OK
445 20.365335  172.168.2.12   172.168.2.100  RTSP  264 PLAY rtsp://172.168.2.100:554/mpeg4/ch01/main/av_stream RTSP/1.0
446 20.369196  172.168.2.100  172.168.2.12   RTSP  182 Reply: RTSP/1.0 200 OK
447 20.445041  172.168.2.100  172.168.2.12   TCP    90 Interleaved channel 0x04, 32 bytes
448 20.445097  172.168.2.12   172.168.2.100  TCP    54 43890 > rtsp [ACK] Seq=604 Ack=878 Win=64820 Len=0
449 20.446035  172.168.2.100  172.168.2.12   RTP  1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14855, Time=2131899382
450 20.446256  172.168.2.100  172.168.2.12   RTP  1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14859, Time=2131899382
451 20.446771  172.168.2.12   172.168.2.100  TCP    54 43890 > rtsp [ACK] Seq=604 Ack=1298 Win=65700 Len=0

Frame 443: 269 bytes on wire (2152 bits), 269 bytes captured (2152 bits)
Ethernet II, Src: a4:1f:72:5d:6e:6d (a4:1f:72:5d:6e:6d), Dst: 8c:e7:48:f1:2b:df (8c:e7:48:f1:2b:df)
Internet Protocol Version 4, Src: 172.168.2.12 (172.168.2.12), Dst: 172.168.2.100 (172.168.2.100)
Transmission Control Protocol, Src Port: 43890 (43890), Dst Port: rtsp (554), Seq: 179, Ack: 536, Len: 215
  Source port: 43890 (43890)
  Destination port: rtsp (554)
  [Stream index: 53]
  Sequence number: 179    (relative sequence number)
  [Next sequence number: 394    (relative sequence number)]
  Acknowledgement number: 536    (relative ack number)
  Header length: 20 bytes
⊞ Flags: 0x18 (PSH, ACK)
  Window size value: 16291
  [Calculated window size: 65164]
  [Window size scaling factor: 4]
⊞ Checksum: 0x5eb2 [validation disabled]
⊞ [SEQ/ACK analysis]
Real Time Streaming Protocol
⊞ Request: SETUP rtsp://172.168.2.100:554/mpeg4/ch01/main/av_stream/trackID=1 RTSP/1.0\r\n
  CSeq:1\r\n
  Authorization:Basic YWRtaW46MTIzNDU=\r\n
  Transport: RTP/AVP/TCP;unicast;interleaved=0-1;ssrc=0
  User-Agent:NKPlayer-1.00.00.081112\r\n
  \r\n
```

Transport: negotiation result of transmission mode. The TCP/RTP mode and unicast transmission are adopted. Interleaved indicates the negotiated channel.

After receiving the setup data, the IPC responds with the data shown in the figure below.

Figure 10



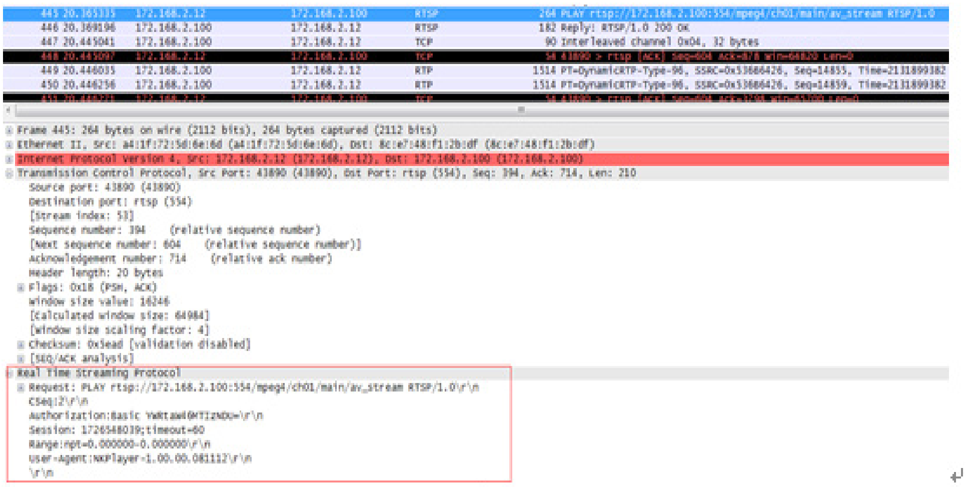
```
444 20.360329  172.168.2.100  172.168.2.12   RTSP  232 Reply: RTSP/1.0 200 OK
445 20.365335  172.168.2.12   172.168.2.100  RTSP  264 PLAY rtsp://172.168.2.100:554/mpeg4/ch01/main/av_stream RTSP/1.0
446 20.369196  172.168.2.100  172.168.2.12   RTSP  182 Reply: RTSP/1.0 200 OK
447 20.445041  172.168.2.100  172.168.2.12   TCP    90 Interleaved channel 0x04, 32 bytes
448 20.445097  172.168.2.12   172.168.2.100  TCP    54 43890 > rtsp [ACK] Seq=604 Ack=878 Win=64820 Len=0
449 20.446035  172.168.2.100  172.168.2.12   RTP  1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14855, Time=2131899382
450 20.446256  172.168.2.100  172.168.2.12   RTP  1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14859, Time=2131899382
451 20.446771  172.168.2.12   172.168.2.100  TCP    54 43890 > rtsp [ACK] Seq=604 Ack=1298 Win=65700 Len=0

⊞ Frame 444: 232 bytes on wire (1856 bits), 232 bytes captured (1856 bits)
⊞ Ethernet II, Src: 8c:e7:48:f1:2b:df (8c:e7:48:f1:2b:df), Dst: a4:1f:72:5d:6e:6d (a4:1f:72:5d:6e:6d)
⊞ Internet Protocol Version 4, Src: 172.168.2.100 (172.168.2.100), Dst: 172.168.2.12 (172.168.2.12)
⊟ Transmission Control Protocol, Src Port: rtsp (554), Dst Port: 43890 (43890), Seq: 536, Ack: 394, Len: 178
    Source port: rtsp (554)
    Destination port: 43890 (43890)
    [Stream index: 53]
    Sequence number: 536    (relative sequence number)
    [Next sequence number: 714    (relative sequence number)]
    Acknowledgement number: 394    (relative ack number)
    Header length: 20 bytes
  ⊞ Flags: 0x18 (PSH, ACK)
    Window size value: 7300
    [Calculated window size: 14600]
    [Window size scaling factor: 2]
  ⊞ Checksum: 0xd1c0 [validation disabled]
  ⊞ [SEQ/ACK analysis]
⊟ Real Time Streaming Protocol
  ⊞ Response: RTSP/1.0 200 OK\r\n
    CSeq: 1\r\n
    Session: 1726548039;timeout=60
    Transport: RTP/AVP/TCP;unicast;interleaved=0-1;ssrc=536b6426;mode="play"
    Date:  Wed, Jan 08 2014 10:52:13 GMT\r\n
    \r\n
```

The most important field is **Transport**, which specifies the video stream transmission mode. mode="play" indicates that the next play negotiation can be conducted.

The third step is the play process.

Figure 11



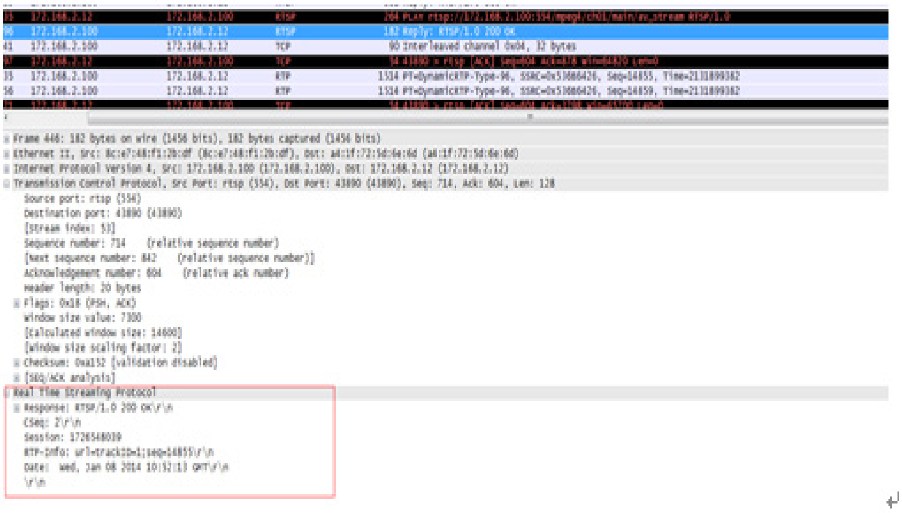
```
445 20.365335   172.168.2.12    172.168.2.100   RTSP   264 PLAY rtsp://172.168.2.100:554/mpeg4/ch01/main/av_stream RTSP/1.0
446 20.369196   172.168.2.100   172.168.2.12    RTSP   182 Reply: RTSP/1.0 200 OK
447 20.445041   172.168.2.100   172.168.2.12    TCP     90 Interleaved channel 0x04, 32 bytes
448 20.445097   172.168.2.12    172.168.2.100   TCP     54 43890 > rtsp [ACK] Seq=604 Ack=878 Win=64820 Len=0
449 20.446035   172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14855, Time=2131899382
450 20.446256   172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14859, Time=2131899382

⊞ Frame 445: 264 bytes on wire (2112 bits), 264 bytes captured (2112 bits)
⊞ Ethernet II, Src: a4:1f:72:5d:6e:6d (a4:1f:72:5d:6e:6d), Dst: 8c:e7:48:f1:2b:df (8c:e7:48:f1:2b:df)
⊞ Internet Protocol Version 4, Src: 172.168.2.12 (172.168.2.12), Dst: 172.168.2.100 (172.168.2.100)
⊟ Transmission Control Protocol, Src Port: 43890 (43890), Dst Port: rtsp (554), Seq: 394, Ack: 714, Len: 210
    Source port: 43890 (43890)
    Destination port: rtsp (554)
    [Stream index: 53]
    Sequence number: 394    (relative sequence number)
    [Next sequence number: 604    (relative sequence number)]
    Acknowledgement number: 714    (relative ack number)
    Header length: 20 bytes
  ⊞ Flags: 0x18 (PSH, ACK)
    Window size value: 16246
    [Calculated window size: 64984]
    [Window size scaling factor: 4]
  ⊞ Checksum: 0x3ead [validation disabled]
  ⊞ [SEQ/ACK analysis]
⊟ Real Time Streaming Protocol
  ⊞ Request: PLAY rtsp://172.168.2.100:554/mpeg4/ch01/main/av_stream RTSP/1.0\r\n
    CSeq:2\r\n
    Authorization:Basic YWRtaW46MTIzNDU=\r\n
    Session: 1726548039;timeout=60
    Range:npt=0.000000-0.000000\r\n
    User-Agent:NKPlayer-1.00.00.081112\r\n
    \r\n
```

Range: specifies the play position and implements the seek operation. This field is not supported in real-time surveillance.

The IPC gives a response, as shown in the figure below.

Figure 12

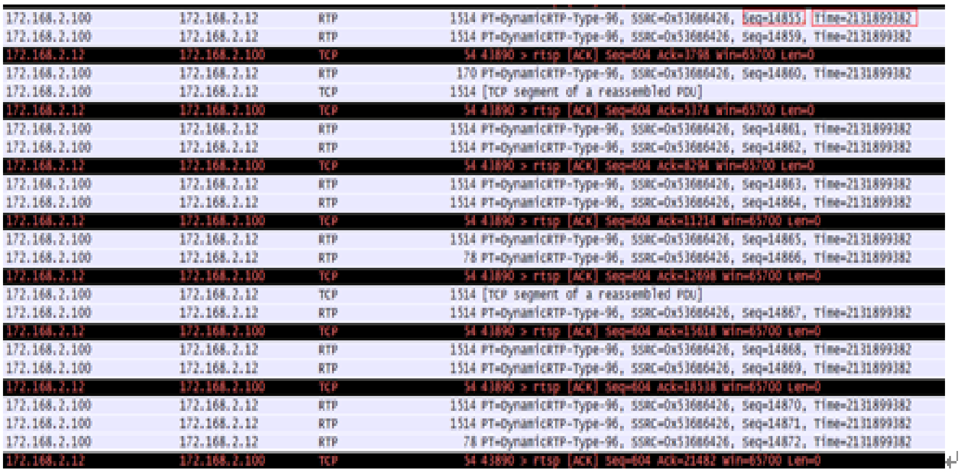


```
35   172.168.2.12    172.168.2.100   RTSP   264 PLAY rtsp://172.168.2.100:554/mpeg4/ch01/main/av_stream RTSP/1.0
96   172.168.2.100   172.168.2.12    RTSP   182 Reply: RTSP/1.0 200 OK
41   172.168.2.100   172.168.2.12    TCP     90 Interleaved channel 0x04, 32 bytes
97   172.168.2.12    172.168.2.100   TCP     54 43890 > rtsp [ACK] Seq=604 Ack=878 Win=64820 Len=0
35   172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14855, Time=2131899382
56   172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14859, Time=2131899382

⊞ Frame 446: 182 bytes on wire (1456 bits), 182 bytes captured (1456 bits)
⊞ Ethernet II, Src: 8c:e7:48:f1:2b:df (8c:e7:48:f1:2b:df), Dst: a4:1f:72:5d:6e:6d (a4:1f:72:5d:6e:6d)
⊞ Internet Protocol Version 4, Src: 172.168.2.100 (172.168.2.100), Dst: 172.168.2.12 (172.168.2.12)
⊟ Transmission Control Protocol, Src Port: rtsp (554), Dst Port: 43890 (43890), Seq: 714, Ack: 604, Len: 128
    Source port: rtsp (554)
    Destination port: 43890 (43890)
    [Stream index: 53]
    Sequence number: 714    (relative sequence number)
    [Next sequence number: 842    (relative sequence number)]
    Acknowledgement number: 604    (relative ack number)
    Header length: 20 bytes
  ⊞ Flags: 0x18 (PSH, ACK)
    Window size value: 7300
    [Calculated window size: 14600]
    [Window size scaling factor: 2]
  ⊞ Checksum: 0xa152 [validation disabled]
  ⊞ [SEQ/ACK analysis]
⊟ Real Time Streaming Protocol
  ⊞ Response: RTSP/1.0 200 OK\r\n
    CSeq: 2\r\n
    Session: 1726548039
    RTP-Info: url=trackID=1;seq=14855\r\n
    Date:  Wed, Jan 08 2014 10:52:13 GMT\r\n
    \r\n
```

After RTSP negotiates the video play mode, it transmits video data in real time. The application layer uses RTP for transmission control regardless of TCP or UDP is adopted. The only difference is that RTP transmits data via UDP and adopts RTCP to reflect the network packet loss and delay. Video vendors need to modify the TCP protocol layer for RTP over TCP data transmission. An implementation solution is available on the public network, and the solution adopted by video vendors is unclear.

Figure 13

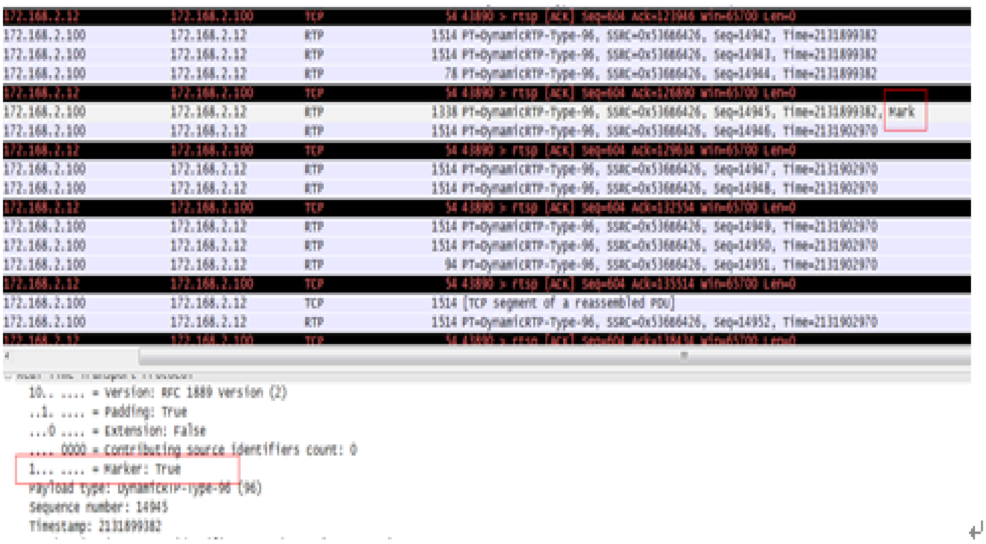


```
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14858, Time=2131899382
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14859, Time=2131899382
172.168.2.12    172.168.2.100   TCP     54 43890 > rtsp [ACK] Seq=604 Ack=1798 Win=65700 Len=0
172.168.2.100   172.168.2.12    RTP    170 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14860, Time=2131899382
172.168.2.100   172.168.2.12    TCP   1514 [TCP segment of a reassembled PDU]
172.168.2.12    172.168.2.100   TCP     54 43890 > rtsp [ACK] Seq=604 Ack=5374 Win=65700 Len=0
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14861, Time=2131899382
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14862, Time=2131899382
172.168.2.12    172.168.2.100   TCP     54 43890 > rtsp [ACK] Seq=604 Ack=8294 Win=65700 Len=0
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14863, Time=2131899382
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14864, Time=2131899382
172.168.2.12    172.168.2.100   TCP     54 43890 > rtsp [ACK] Seq=604 Ack=11214 Win=65700 Len=0
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14865, Time=2131899382
172.168.2.100   172.168.2.12    RTP     78 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14866, Time=2131899382
172.168.2.12    172.168.2.100   TCP     54 43890 > rtsp [ACK] Seq=604 Ack=12698 Win=65700 Len=0
172.168.2.100   172.168.2.12    TCP   1514 [TCP segment of a reassembled PDU]
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14867, Time=2131899382
172.168.2.12    172.168.2.100   TCP     54 43890 > rtsp [ACK] Seq=604 Ack=15618 Win=65700 Len=0
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14868, Time=2131899382
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14869, Time=2131899382
172.168.2.12    172.168.2.100   TCP     54 43890 > rtsp [ACK] Seq=604 Ack=18538 Win=65700 Len=0
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14870, Time=2131899382
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14871, Time=2131899382
172.168.2.100   172.168.2.12    RTP     78 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14872, Time=2131899382
172.168.2.12    172.168.2.100   TCP     54 43890 > rtsp [ACK] Seq=604 Ack=21482 Win=65700 Len=0
```

Seq: sequence number, used to reassemble video frames.

Time: timestamp. Frames with the same timestamp indicates the same frames.

Figure 14



```
172.168.2.12    172.168.2.100   TCP     54 43890 > rtsp [ACK] Seq=604 Ack=123946 Win=65700 Len=0
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14942, Time=2131899382
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14943, Time=2131899382
172.168.2.100   172.168.2.12    RTP     78 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14944, Time=2131899382
172.168.2.12    172.168.2.100   TCP     54 43890 > rtsp [ACK] Seq=604 Ack=126890 Win=65700 Len=0
172.168.2.100   172.168.2.12    RTP   1338 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14945, Time=2131899382, Mark
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14946, Time=2131902970
172.168.2.12    172.168.2.100   TCP     54 43890 > rtsp [ACK] Seq=604 Ack=129634 Win=65700 Len=0
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14947, Time=2131902970
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14948, Time=2131902970
172.168.2.12    172.168.2.100   TCP     54 43890 > rtsp [ACK] Seq=604 Ack=132554 Win=65700 Len=0
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14949, Time=2131902970
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14950, Time=2131902970
172.168.2.100   172.168.2.12    RTP     94 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14951, Time=2131902970
172.168.2.12    172.168.2.100   TCP     54 43890 > rtsp [ACK] Seq=604 Ack=135514 Win=65700 Len=0
172.168.2.100   172.168.2.12    TCP   1514 [TCP segment of a reassembled PDU]
172.168.2.100   172.168.2.12    RTP   1514 PT=DynamicRTP-Type-96, SSRC=0x53686426, Seq=14952, Time=2131902970
172.168.2.12    172.168.2.100   TCP     54 43890 > rtsp [ACK] Seq=604 Ack=138434 Win=65700 Len=0

Real-Time Transport Protocol
    10.. .... = Version: RFC 1889 Version (2)
    ..1. .... = Padding: True
    ...0 .... = Extension: False
    .... 0000 = Contributing source identifiers count: 0
    1... .... = Marker: True
    Payload type: DynamicRTP-Type-96 (96)
    Sequence number: 14945
    Timestamp: 2131899382
```

Marker: indicates that one frame has ended.

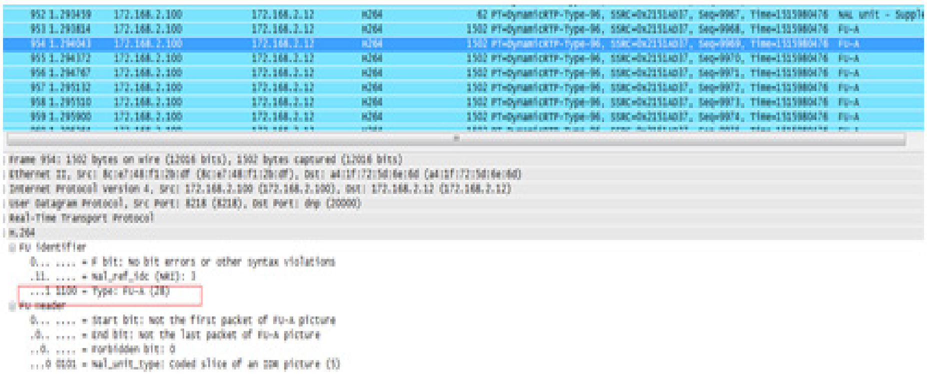
The RFC 3984 specifies the bearer modes of H.264 over RTP, which are shown in the figure below.

Figure 15

Type	Packet	Single NAL Unit Mode	Non~Interleaved Mode	Interleaved Mode
0	undefined	ig	ig	ig
1-23	NAL unit	yes	yes	no
24	STAP-A	no	yes	no
25	STAP-B	no	no	yes
26	MTAP16	no	no	yes
27	ITTAP24	no	no	yes
28	FU-A	no	yes	yes
29	FU-B	no	no	yes
30-31	undefined	ig	ig	ig

When the size of a video frame is greater than the Maximum Transmission Unit (MTU), Fragmentation Unit Type A (FU-A) (fragmented transmission) is usually adopted. The FU-A mode is also used for IP surveillance, as shown in the figure below.

Figure 16



```
952 1.293439   172.168.2.100   172.168.2.12   H264     62 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=9967, Time=1315980476  NAL unit - Supple
953 1.293814   172.168.2.100   172.168.2.12   H264   1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=9968, Time=1315980476  FU-A
954 1.294043   172.168.2.100   172.168.2.12   H264   1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=9969, Time=1315980476  FU-A
955 1.294372   172.168.2.100   172.168.2.12   H264   1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=9970, Time=1315980476  FU-A
956 1.294767   172.168.2.100   172.168.2.12   H264   1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=9971, Time=1315980476  FU-A
957 1.295132   172.168.2.100   172.168.2.12   H264   1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=9972, Time=1315980476  FU-A
958 1.295510   172.168.2.100   172.168.2.12   H264   1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=9973, Time=1315980476  FU-A
959 1.295900   172.168.2.100   172.168.2.12   H264   1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=9974, Time=1315980476  FU-A

Frame 954: 1502 bytes on wire (12016 bits), 1502 bytes captured (12016 bits)
Ethernet II, Src: 8c:e7:48:f1:2b:df (8c:e7:48:f1:2b:df), Dst: a4:1f:72:5d:6e:6d (a4:1f:72:5d:6e:6d)
Internet Protocol Version 4, Src: 172.168.2.100 (172.168.2.100), Dst: 172.168.2.12 (172.168.2.12)
User Datagram Protocol, Src Port: 8218 (8218), Dst Port: dnp (20000)
Real-Time Transport Protocol
H.264
  FU identifier
    0... .... = F bit: No bit errors or other syntax violations
    .11. .... = Nal_ref_idc (NRI): 3
    ...1 1100 = Type: FU-A (28)
  FU Header
    0... .... = Start bit: Not the first packet of FU-A picture
    .0.. .... = End bit: Not the last packet of FU-A picture
    ..0. .... = Forbidden bit: 0
    ...0 0101 = Nal_unit_type: Coded slice of an IDR picture (5)
```

FA-U uses the following format for fragmentation. H.264 fragmentation is described using FU-A as follows:

1. FU indicator of the first FU-A packet: The F bit should be the F bit in the current NALU header, NRI should be the NRI of the current NALU header, and Type is 28, indicating that the packet is a FU-A packet. The FU header is generated as follows: S = 1, E = 0, R = 0, and Type is Type in the NALU header.

2. The FU indicator of the subsequent N FU-A packets is the same as that of the first FU-A packet. If an FU-A packet is not the last packet, the FU header should be as follows: S = 0, E = 0, R = 0, and Type is Type in the NALU header.

3. The FU header of the last FU-A packet should be as follows: S = 0, E = 1, R = 0, and Type is Type in the NALU header.

Figure 17

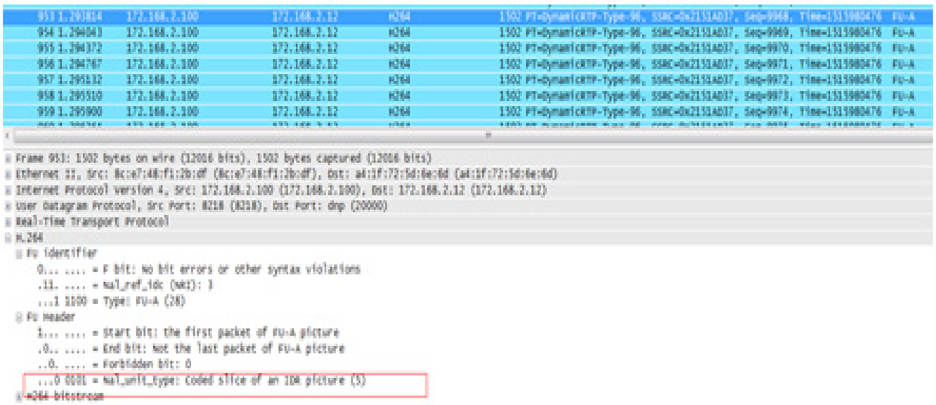
```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| FU indicator         FU header |                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               |
|                                                               |
|                           FU payload                          |
|                                                               |
|                               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                               : . . . OPTIONAL RTP padding    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

I frames and P frames can be differentiated by NAL_unit_type.

nal_unit_type	Content of NAL Unit and RBSP Syntax Structure	
1	Coded slice of a non-Instantaneous Decoding Refresh (IDR) picture	P frame
5	Coded slice of an IDR picture	I frame
6	Supplemental Enhancement Information (SEI)	
7	Sequence parameter set	
8	Picture parameter set	
9	Access unit delimiter	

The figure below shows an example of an I frame.

Figure 18

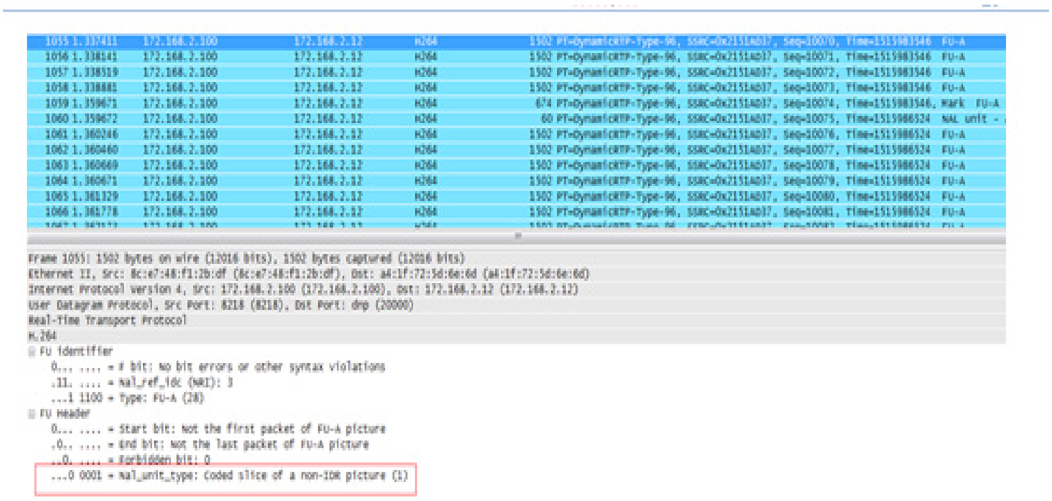


```
953 1.293814  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=9968, Time=1513980476  FU-A
954 1.294043  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=9969, Time=1513980476  FU-A
955 1.294372  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=9970, Time=1513980476  FU-A
956 1.294767  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=9971, Time=1513980476  FU-A
957 1.295132  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=9972, Time=1513980476  FU-A
958 1.295510  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=9973, Time=1513980476  FU-A
959 1.295900  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=9974, Time=1513980476  FU-A

Frame 953: 1502 bytes on wire (12016 bits), 1502 bytes captured (12016 bits)
Ethernet II, Src: 8c:e7:48:f1:2b:df (8c:e7:48:f1:2b:df), Dst: a4:1f:72:5d:6e:6d (a4:1f:72:5d:6e:6d)
Internet Protocol Version 4, Src: 172.168.2.100 (172.168.2.100), Dst: 172.168.2.12 (172.168.2.12)
User Datagram Protocol, Src Port: 8218 (8218), Dst Port: dnp (20000)
Real-Time Transport Protocol
H.264
  FU identifier
    0... .... = F bit: No bit errors or other syntax violations
    .11. .... = Nal_ref_idc (NRI): 3
    ...1 1100 = Type: FU-A (28)
  FU Header
    1... .... = Start bit: the first packet of FU-A picture
    .0.. .... = End bit: Not the last packet of FU-A picture
    ..0. .... = Forbidden bit: 0
    ...0 0101 = Nal_unit_type: Coded slice of an IDR picture (5)
  H264 bitstream
```

The figure below shows an example of a P frame.

Figure 19

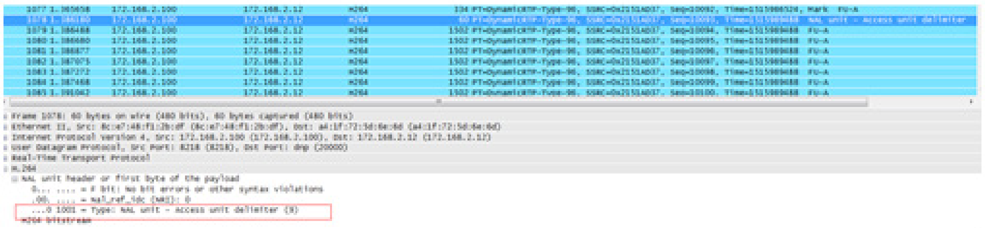


```
1055 1.337411  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10070, Time=1515983546  FU-A
1056 1.338141  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10071, Time=1515983546  FU-A
1057 1.338519  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10072, Time=1515983546  FU-A
1058 1.338881  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10073, Time=1515983546  FU-A
1059 1.359671  172.168.2.100  172.168.2.12  H264   674 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10074, Time=1515983546, Mark  FU-A
1060 1.359672  172.168.2.100  172.168.2.12  H264    60 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10075, Time=1515986524  NAL unit -
1061 1.360246  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10076, Time=1515986524  FU-A
1062 1.360460  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10077, Time=1515986524  FU-A
1063 1.360669  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10078, Time=1515986524  FU-A
1064 1.360671  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10079, Time=1515986524  FU-A
1065 1.361329  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10080, Time=1515986524  FU-A
1066 1.361778  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10081, Time=1515986524  FU-A

Frame 1055: 1502 bytes on wire (12016 bits), 1502 bytes captured (12016 bits)
Ethernet II, Src: 8c:e7:48:f1:2b:df (8c:e7:48:f1:2b:df), Dst: a4:1f:72:5d:6e:6d (a4:1f:72:5d:6e:6d)
Internet Protocol Version 4, Src: 172.168.2.100 (172.168.2.100), Dst: 172.168.2.12 (172.168.2.12)
User Datagram Protocol, Src Port: 8218 (8218), Dst Port: dnp (20000)
Real-Time Transport Protocol
H.264
⊟ FU identifier
    0... .... = F bit: No bit errors or other syntax violations
    .11. .... = Nal_ref_idc (NRI): 3
    ...1 1100 = Type: FU-A (28)
⊟ FU Header
    0... .... = Start bit: Not the first packet of FU-A picture
    .0.. .... = End bit: Not the last packet of FU-A picture
    ..0. .... = Forbidden bit: 0
    ...0 0001 = Nal_unit_type: Coded slice of a non-IDR picture (1)
```

A packet shown in the figure below is generated at the fragmentation position of each video frame.

Figure 20

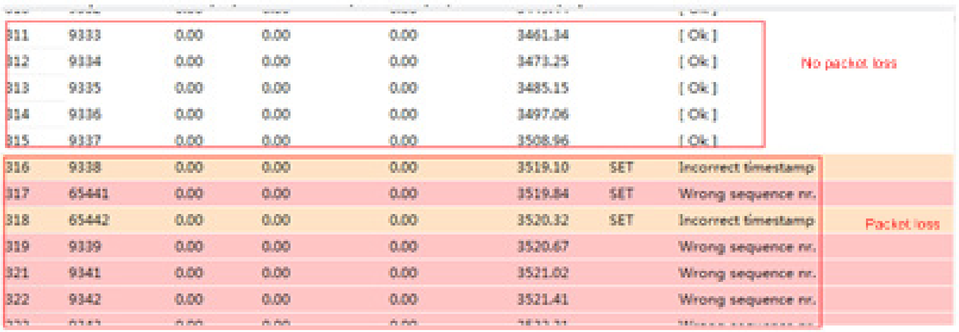


```
1077 1.385658  172.168.2.100  172.168.2.12  H264   334 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10092, Time=1515986524, Mark  FU-A
1078 1.386180  172.168.2.100  172.168.2.12  H264    60 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10093, Time=1515989488  NAL unit - Access unit delimiter
1079 1.386488  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10094, Time=1515989488  FU-A
1080 1.386680  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10095, Time=1515989488  FU-A
1081 1.386877  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10096, Time=1515989488  FU-A
1082 1.387075  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10097, Time=1515989488  FU-A
1083 1.387272  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10098, Time=1515989488  FU-A
1084 1.387468  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10099, Time=1515989488  FU-A
1085 1.391042  172.168.2.100  172.168.2.12  H264  1502 PT=DynamicRTP-Type-96, SSRC=0x2151AD37, Seq=10100, Time=1515989488  FU-A

⊞ Frame 1078: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
⊞ Ethernet II, Src: 8c:e7:48:f1:2b:df (8c:e7:48:f1:2b:df), Dst: a4:1f:72:5d:6e:6d (a4:1f:72:5d:6e:6d)
⊞ Internet Protocol Version 4, Src: 172.168.2.100 (172.168.2.100), Dst: 172.168.2.12 (172.168.2.12)
⊞ User Datagram Protocol, Src Port: 8218 (8218), Dst Port: dnp (20000)
⊞ Real-Time Transport Protocol
⊟ H.264
  ⊟ NAL unit header or first byte of the payload
      0... .... = F bit: No bit errors or other syntax violations
      .00. .... = Nal_ref_idc (NRI): 0
      ...0 1001 = Type: NAL unit - Access unit delimiter (9)
    H264 bitstream
```

In normal cases, the main control PC reassembles video frames after receiving RTP packets. If a frame loss occurs during transmission, reassembling video frames may fail.

Figure 21



311	9333	0.00	0.00	0.00	3461.34		[Ok]
312	9334	0.00	0.00	0.00	3473.25		[Ok]
313	9335	0.00	0.00	0.00	3485.15		[Ok]
314	9336	0.00	0.00	0.00	3497.06		[Ok]
315	9337	0.00	0.00	0.00	3508.96		[Ok]
316	9338	0.00	0.00	0.00	3519.10	SET	Incorrect timestamp
317	65441	0.00	0.00	0.00	3519.84	SET	Wrong sequence nr.
318	65442	0.00	0.00	0.00	3520.32	SET	Incorrect timestamp
319	9339	0.00	0.00	0.00	3520.67		Wrong sequence nr.
321	9341	0.00	0.00	0.00	3521.02		Wrong sequence nr.
322	9342	0.00	0.00	0.00	3521.41		Wrong sequence nr.

No packet loss

Packet loss

I frames, are key frames. It is necessary to ensure reliable transmission of I frames. One I frame affects and determines the decoding of several subsequent P frames. The loss of one I frame will result in the play failure of the video between two I frames or sharp video quality deterioration. P frames are non-key frames. The loss of a P frame is insignificant to the video quality. In general, P frames are transmitted in best-effort manner and will not be retransmitted upon errors.

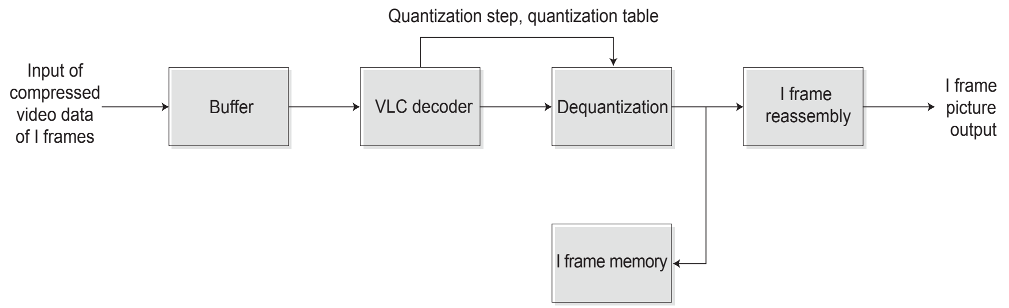
The loss of I frames will cause pixelation on video pictures. P frames change based on the previous I frame. The loss of an I frame will affect the quality of subsequent P frame pictures. Normal pictures are restored only after the next I frame is received. P frames carry less information than I frames. The impact of P frame loss on the videos is temporary, for example, pixilation or a color exception occurs on an area, the motion is abnormal, or the picture flutters.

• Processing at the Receive End

The receive end decodes the received video frames as follows:

* **Decoding I frames**

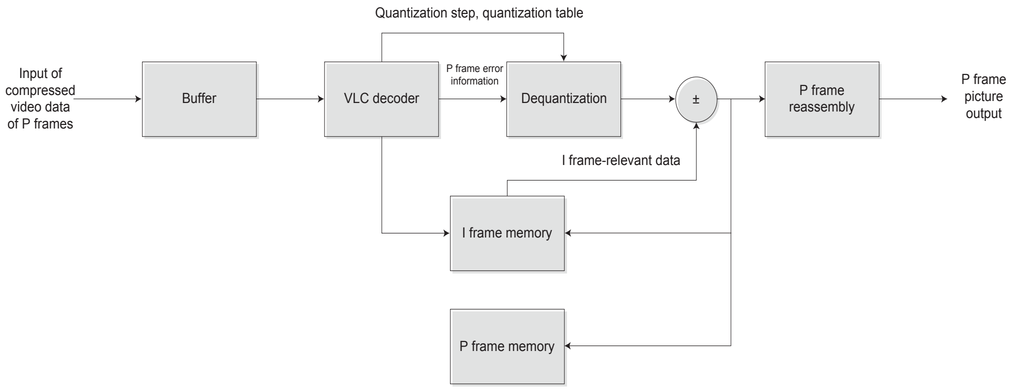
Figure 22



When compressed data of I frame pictures is input, the buffer stores the data and the VLC decoder extracts the quantization step and quantization table to control dequantization and obtain the frequency coefficient. Then, the discrete cosine converter processes the data to generate spatial domain data, which is transmitted through two channels. In one channel, the frame arrangement sequence prior to encoding is restored by means of I frame reassembly for frame output. In the other channel, the data is transferred to the I frame memory, which will serve as the benchmark in the decoding of subsequent P frames or B frames.

* **Decoding P frames**

Figure 23



When compressed data of P frame pictures is input, the buffer stores the data and the VLC decoder extracts the quantization step and quantization table to control dequantization. The VLC decoder also outputs a motion vector and transfers it to the I frame memory, so as to obtain I frame-relevant data. The VLC decoder outputs P frame error data and conducts dequantization and discrete cosine conversion to obtain predicted error data. The predicted error data is added together with the I frame-relevant data in the adder to obtain P frame pictures, which are transmitted through two channels. In one channel, the P frame arrangement sequence prior to encoding is restored by means of P frame reassembly for frame output. In the other channel, the data transferred to the P frame memory, which will serve as the benchmark in the decoding of subsequent B frames.

Video streams received by the receive end involve the bitstream management mechanism in addition to decoding.

* **Bitstream receiving: The receive end not only receives bitstreams sent from the transmit end but also analyzes and decodes the bitstreams, which is a time-consuming process. If the transmit end sends bitstreams very rapidly and the receive end performs this process in sequence, the receive end may fail to receive complete packets from the transmit end and packet loss may occur. As a result, major errors such as decoding error, video play failure, and program breakdown will occur. The concurrent processing mechanism is adopted to address this problem. This mechanism not only implements fast data receiving but also ensures the correct sequencing of received packets. It separates receiving from analysis and decoding. The receive end temporarily stores received data in the buffer, and then receives the next packet data, with no need to wait for the completion of analysis and decoding. This greatly improves the receiving efficiency and prevents packet loss.**

* **Analysis and decoding of video data: More than one packet is received at a time because of the concurrent mechanism. Therefore, how to properly handle received data is a great difficulty. It is necessary to ensure the sequence of data packets and only one packet can be processed each time, which involves the collaboration between threads. The consumer-producer thread collaboration mode is adopted for processing. Packets are obtained from the data storage buffer in sequence, analyzed, and then put into another buffer. Then, the decoding program is notified to obtain data from this buffer for decoding. Subsequently, the video data analysis program enters the waiting state. After decoding is completed, the video data analysis program is notified to analyze video data, and the decoding program enters the waiting state. The two programs enter the execution and waiting states alternately.**

* **Multi-level buffer mechanism: The involved buffers are as follows:**

1. Buffer for storing received data: IPCs send real-time bitstreams constantly and a larger amount of data is received due to the concurrent mechanism. Not all data can be processed immediately, and a buffer is necessary.

2. Buffer between the receive end and processing end: Network instability will bring about different data receiving rates, which will directly cause decoding instability and video play discontinuity. Therefore, a buffer is required to smooth data. This buffer needs to store a large number of received bitstreams and provide data for video data analysis. It involves a write/read-in and read-out process. Therefore, a first in first out queue container is used to function as the buffer.

3. Buffer between analysis and decoding: Though the data amount is not large, the data acquisition speed directly affects the decoding speed. Therefore, an efficient buffer is required.

4. (Buffer between decoding and play: This buffer aims at making video play continuous and stable, and is intended for picture display and video picture processing, such as smoothing and filtering.

Traffic Characteristics

• Basic Parameters

According to the technical principle of the video stream technology in the foregoing chapter, the following parameters determine the video traffic and distribution characteristics:

* **Bit rate**

* **Size of I frames and P frames**

* **I frame interval**

* **Frame rate**

The bit rate determines the upper limit of the average traffic of video streams, or in other words, the theoretical maximum required bandwidth. It is the sum of the products of the latter three parameters. Bit rate includes Constant Bit Rate (CBR) and Variable Bit Rate (VBR).

The latter three parameters determine the traffic balance, the distribution of video streams and traffic sizes at different time points.

• CBR and VBR

CBR

When CBR is adopted, the bit rate of video streams is basically constant and is close to the target bit rate. The CBR implementation is simple, the computation load is light, the encoding time is short, and the decoding algorithm is simple. In complex scenarios, however, pixilation may occur due to insufficient bit rate; in simple scenarios, the encoding space cannot be fully utilized.

VBR

When VBR is adopted, the frame-by-frame scanning is performed to raise the bit rate for complex scenarios with abundant motions, and lower the bit rate for static scenarios or scenarios with few motions, and it is ensured that the equalized overall bit rate does not exceed the preset average bit rate. Therefore, VBR is advantageous in high picture quality but requires heavy computation load and long compression time, and also raises high requirements for decoding.

In summary, the bit rate of video streams at CBR fluctuates slightly and the CBR implementation is simple. Pixelation or stuttering may occur if the bit rate is improper in complex scenarios. The VBR uses different bit rates based on the scenario. The bit rate of video streams fluctuates greatly, the overall picture quality is good, but the VBR implementation is complex.

• Characteristics Analysis

Original Traffic Waveform

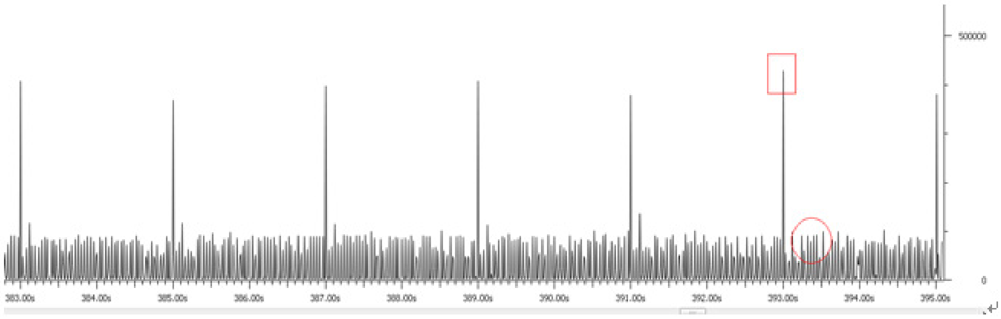
Physical ports of an IPC automatically negotiate to work at 100 Mbps. On the web page, manually configure UDP and VBR, set the bit rate to 2 Mbps, I frame interval to 50, and frame rate to 25 fps for Dahua HF5200, and shoot static pictures. Theoretically, the maximum average traffic of video streams is 2 Mbps, one I frame and 49 P frames compose one GOP, and it takes 2s to transmit one GOP.

The figure below shows the video traffic distribution captured by Wireshark.

The horizontal axis indicates time in seconds while the vertical axis indicate traffic in bits. The sampling frequency of the Wireshark is 10 ms, that is, the burrs on the waveform indicate the total traffic sent by the IPC every 10 ms.

The traffic shown in the figure is unstable. The highly protruding burrs indicate I frames (marked in a red rectangle), and the low protruding burrs between I frames indicate P frames (marked in a red circle). The interval between I frames is about 2s, and there are about 49 P frames between two I frames, which meet expected results.

Figure 24

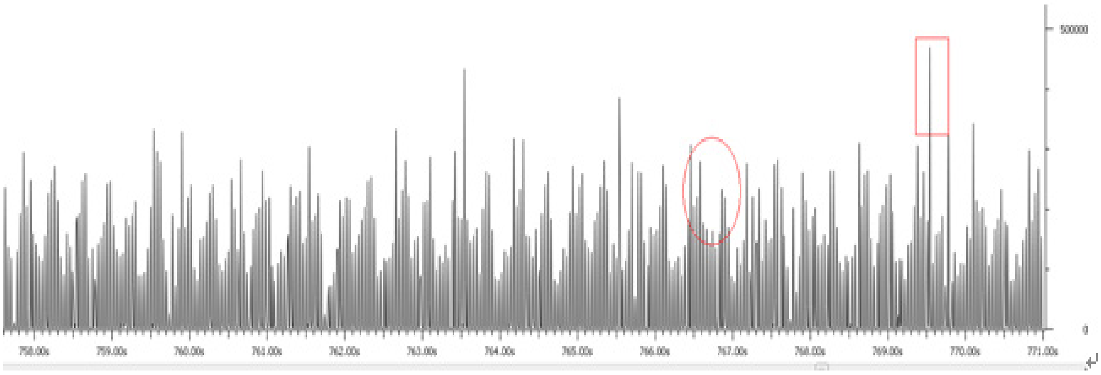


The UDP transmission mode is set for this IPC. No flow control mechanism is available according to the basic principle of UDP. Therefore, it can be deemed that the waveform reflects the original behavior characteristics of the IPC, that is, the IPC sends out I frames or P frames in best-effort manner once they are generated, and the traffic bursts each time I frames or P frames are sent, and the bursts are manifested with burrs on the waveform. The waveform above shows that the value of the highest burr is about 433 kbit (marked in the red rectangle), which is based on the sampling frequency of 10 ms, that is, 43.3 Mbit/s. Under current settings, when the IPC ports automatically negotiate to work at the bandwidth of 100 Mbps and send I frames and P frames in best-effort manner, the maximum instantaneous burst traffic can exceed 40 Mbps. The original video stream is uneven without traffic shaping.

Waveforms of Motion Pictures

The original waveform described in 2.3.1 is based on static pictures. Under the same settings, shoot a continuously shaking motion picture. The figure below shows the waveform.

Figure 25



The difference between the two waveforms lies in the size of P frame burrs. P frames do not contain information of the entire pictures but only differences from the previous frames. In the motion pictures, however, there are a lot of changes and the size of P frames becomes larger. I frames contain information about the entire pictures at a time point regardless of static pictures or motion pictures. The size of I frames is not much different.

In conclusion, when shooting motion pictures, IPCs generate larger P frames, leading to higher P frame bursts.

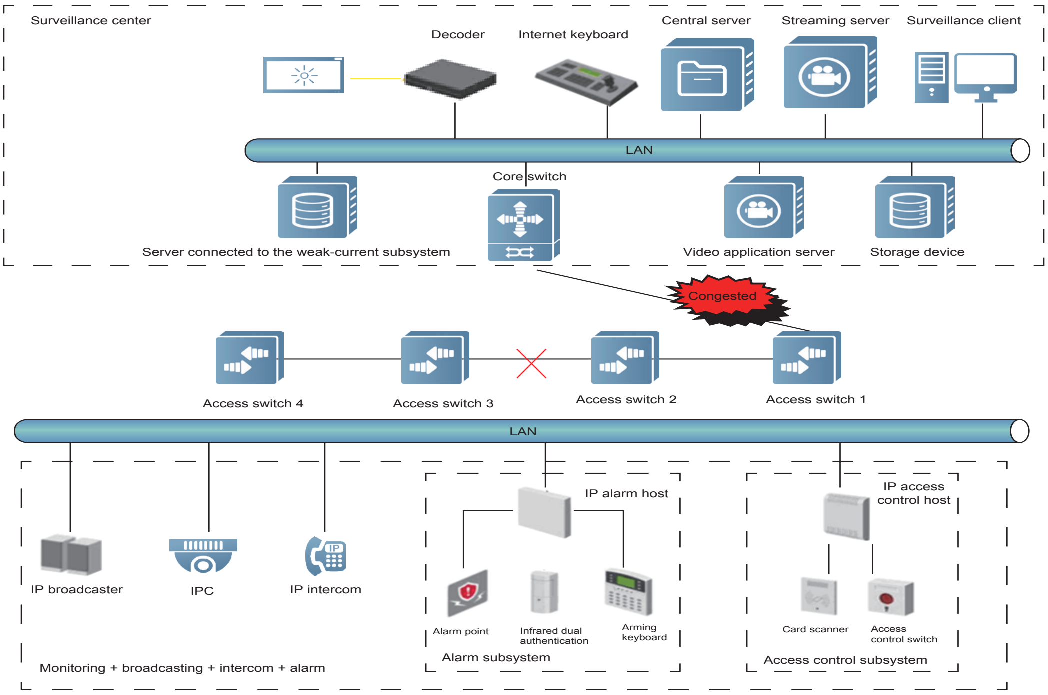
Waveforms at CBR

Under the same settings, capture waveforms of one motion picture and one static picture at CBR. The waveform patterns and conclusions are basically the same as those of waveforms at VBR.

Typical Applications of IP Surveillance

• Centralized Storage Application of a Single Surveillance Center

Figure 26



The figure above shows the common IP surveillance system, which is commonly applied in the security system of an independent building. This IP surveillance system is applicable to medium- and small-scale surveillance projects. A separate surveillance network is connected to all security-relevant subsystems, video data is stored only in the sole surveillance center, and the video on-demand client is also deployed in the surveillance center. The following problems may arise in this scenario:

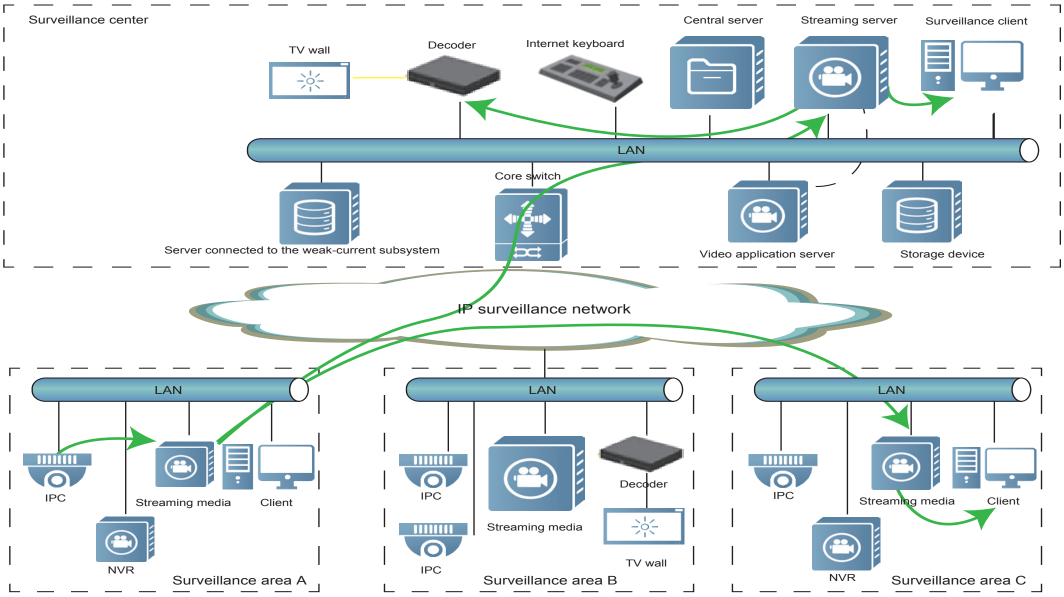
*** Different from the production network, on the video dedicated network, users may have smaller investment and will not use the dual-core dual uplink networking mode. Users may pose high requirements for the video recording reliability, adopt the chain networking mode to reduce investment in optical fibers. In this case, if the link between Access Switch 2 and Access Switch 3 is interrupted, IP surveillance information in devices connected to Access Switch 3 and Access Switch 4 will be lost.**

*** If chain networking is adopted, the link between Access Switch 1 and the core switch may be congested due to abnormal traffic, resulting in the loss of video files. For example, if IPCs support the video buffering function, when Access Switch 1 drops out of the network, all IPCs on the chain network fail to store video data to the rear end. If the IPCs are equipped with SD cards, video data generated during network interruption is stored to the SD cards, and then stored to the rear end after the network is restored. If Access Switch 1 goes online again, the burst traffic doubles, increasing bandwidth pressure temporarily.**

*** Centralized storage imposes great pressure on the backbone network between access devices and core devices. There are also preview streams and storage streams on the network, and no problem occurs if the bandwidth is sufficient. However, video information traffic may burst. If burst traffic is superimposed onto normal burst traffic (for example, much I frame information from IPCs), the bandwidth becomes insufficient and packet loss will occur, affecting the video quality.**

• Distributed Storage Application of Multiple Surveillance Centers

Figure 27



When surveillance areas belong to a hierarchical structure, or considerable surveillance areas exist and need to be further divided, a multi-level surveillance system arises. Video data is stored locally in multi-level surveillance application. The real-time video preview may occur in all surveillance areas. Streaming servers are required to distribute and duplicate video streams due to limitations on the bandwidth and access quantity of IPCs or storage devices. In Surveillance Area A in the figure above, the streaming server duplicates one video stream into two streams for distribution. If video sources in Surveillance Area A increase or stream acquisition clients in other areas increase, the following problems may arise:

*** The limited forwarding performance of the streaming server cannot meet constantly increasing requirements of stream acquisition clients and becomes a system bottleneck.**

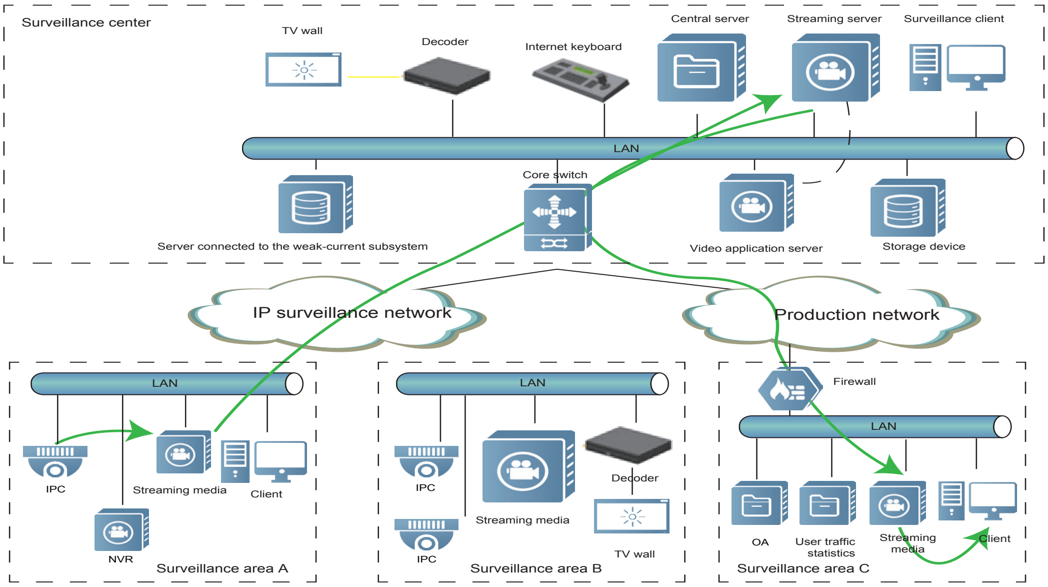
*** Delay increases when video streams are forwarded by the streaming server.**

*** When streaming media are added to eliminate the system bottleneck, the total investment of the surveillance system soars.**

*** Multiple preview streams impose great pressure on the backbone network between access devices and core devices. No problem occurs if the bandwidth is sufficient. However, video information traffic may burst. If burst traffic is superimposed onto normal burst traffic (for example, much I frame information from IPCs), the bandwidth becomes insufficient and packet loss will occur, affecting the video quality.**

• Application of Dedicated IP Surveillance Network as a Subnet

Figure 28



In some application scenarios, the dedicated surveillance network serves as a subnet of the production network, to provide video pictures or statistical data for the production network. If the IP surveillance network and production network belong to the same subnet, the following problems may arise:

*** The backbone network between access devices and core devices is under great pressure. No problem occurs if the bandwidth is sufficient. However, video information traffic may burst. If burst traffic is superimposed onto normal burst traffic (for example, much I frame information from IPCs), the bandwidth becomes insufficient and packet loss will occur, affecting the video quality.**

*** The traffic of the surveillance network needs to be distinguished from that of the production network for processing in the core network. Otherwise, service functions will be affected.**

Transmission Analysis

• Major Problems

According to video network scenario analysis in the foregoing chapter, video streams are characterized by traffic burst. If multiple video streams are superimposed onto burst video streams, the video stream burst is severer, which not only exceeds the egress bandwidth but also is beyond the buffer capacity of the switch, leading to packet loss. In actual network environments, congestion may occur and bandwidth may be affected, video stream packet loss is more prone to occur.

• Switch Buffer Analysis

A large amount of test data shows that the toughest environment and longest traffic burst duration occur at the time that an I frame and the first P frame are sent (an I frame is large and cannot be sent within one frame interval; as a result, the I frame and a P frame are sent out continuously). The size of an I frame is about 48-85 Kbytes and the size of a P frame is about 10–35 Kbytes. The heaviest continuous burst traffic is 120 Kbytes. (Set the frame rate to 25 fps, bit rate to 2 Mbps, resolution to 720P, and GOP to 50, select CBR and TCP transmission mode for an IPC, and shoot a long corridor, in which personnel move around randomly.)

The buffer capacity of the switch is theoretically analyzed as follows: Use the access switch S26I as an example. The buffer capacity of the S26I is 1 Mbyte. In basic application scenarios, the port rate of the downlink IPC connected to the S26I is 100 Mbps, and the port rate of the uplink core switch is 1 Gbps.

In the worst case, the burst traffic of I frames and the first P frames of all IPCs are superimposed. The 1Gbps bandwidth of the uplink port is just adequate for the ten 100Mbps downlink ports (the traffic is calculated based on port bandwidth here and it is deemed that IPCs can consume 100 Mbps bandwidth). If there is one more IPC stream, the ingress traffic exceeds the egress bandwidth theoretically, video streams cannot be forwarded in a timely manner and have to be buffered in the switch.

The burst traffic of one IPC stream is 120 Kbyte. The buffer capacity of the S26I is 1 Mbyte, indicating that the burst traffic of 8.5 IPC streams can be buffered. That is, the switch is capable of processing burst traffic of 8 IPC streams at most theoretically, and packet loss may occur when there is another one or more IPC streams.

Note that the buffer capacity of a switch refers to the maximum buffer capacity of the entire switch or a switching chip. A targeted (chip-dependent) buffer resource allocation and management plan is adopted in the switch design and implementation, in an attempt to adapt to most network traffic scenarios. The overall buffer allocation philosophy is identical except for some details. That is, all ports can contend for buffer resources fairly, and it is impossible that one congested port depletes all buffer resources. On this basis, consideration is taken for some common application or test scenarios, for example, high-speed input to low-speed output, multi-port input to single-port output (flow control), single-port input to multi-port output (HOL), full mesh & backbone test models. Base on these scenarios, an appropriate buffer allocation policy is worked out finally. Take Ruijie S26I 10.4(3b16) as an example. The maximum buffer capacity of its switching chip is 1 Mbyte. One 100M port can consume a maximum of about 0.066 Mbytes buffer space and one 1G port can consume a maximum of about 0.198 Mbytes buffer space. The preceding theoretical analysis result that the S26I is capable of processing burst traffic of 8 IPC streams in the worst case, is also based on extreme conditions (that is, one uplink port depletes all buffer resources).

• Switch Buffer Test

This section uses Dahua HF5200 IPC to test the burst traffic of how many IPC streams can be stored in the buffer of the S26I.

Get ready eight IPCs, set the bit rate to 2 Mbps, I frame interval to 50, resolution to 720P, and select UDP transmission for the IPCs. The required egress bandwidth of the uplink port is 16 Mbps theoretically. A transmission model of sending traffic from eight 100M ports to one 1G port is formed ultimately.

Run the rate-limit command on the S26I to gradually reduce the egress bandwidth of the uplink port and observe the packet loss status on the switch. It is found that packet loss occurs on the switch when the egress bandwidth is reduced to 19 Mbps. Therefore, a critical egress bandwidth value is about 20 Mbps. The uplink port still has 980Mbps idle bandwidth (buffering incapable) after the 20Mbps bandwidth is subtracted. IPCs access the surveillance network at the rate of100 Mbps. Therefore, the switch can still handle the traffic even if ten IPCs are added. In the preceding test, the burst traffic of one IPC stream is controlled around 50 Mbps. It can be inferred that the S2628I switch can cope with the traffic even if 24 IPCs (24 x 100M downlink ports and 1 x 1000M uplink port) operating at the bit rate of 2 Mbps are connected, without causing packet loss.

This is the case that the bit rate is only 2 Mbps and each IPC generates only one data stream. A higher traffic burst superimposition probability and heavier burst traffic will be incurred if the bit rate is higher, or one IPC generates multiple data streams, or the I frame interval is narrowed, or a switch has more ports such as the S2652I. Consequently, the switch load will increase and congestion and packet loss will occur on the switch easily. The buffer capacity of the switch is fixed, and it cannot handle such great variances.

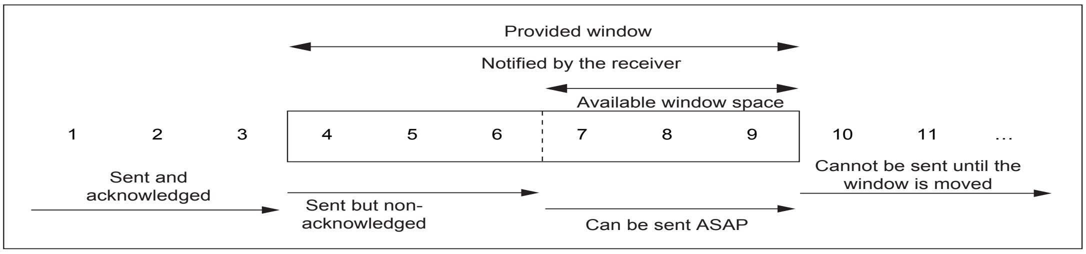
Therefore, it is inferred that the switch buffer capacity is insufficient to handle the superimposed traffic of I frames and the first P frames of multiple video streams, and packet loss and even stuttering may occur. In actual cases, the sum of average bandwidth of video streams is lower than the egress bandwidth of the uplink port.

• TCP Advantages

The video stream analysis above is based on the UDP transmission mode. UDP is an unreliable transmission protocol. Packet loss occurs if the bandwidth is affected due to traffic burst or network congestion. TCP is a reliable transmission protocol, and has the flow control and retransmission mechanisms. It is advantageous in the handling of congestion or packet loss.

The figure below shows the basic principle of TCP flow control.

Figure 29



In the figure, bytes are numbered 1 to 11. The receiver notifies the sender that the window size is 6 bytes. The window is "Provided window" in the figure.

The window covers the area from Byte 4 to Byte 9, indicating that the receiver has acknowledged the receiving of Byte 1 to Byte 3.

Then, the sender continues to send data from Byte 4, and the data from Byte 4 to Byte 6 has been sent but not acknowledged by the receiver. The window size is 6 bytes. Only 3 bytes are sent, and there is idle space for another 3 bytes. The data from Byte 7 to Byte 9 can also be sent, but Byte 10 and subsequent data cannot be sent due to insufficient window size.

When the receiver acknowledges the receiving of Byte 4, there is idle space for another one byte and Byte 10 can be sent.

This window is intended to prevent the sender from sending data packets ceaselessly when it is not confirmed whether the receiver has received data. In this way, the maximum burst traffic of the sender is restricted by the window, thereby achieving flow control. In the example, the burst traffic of the sender is only 6 bytes.

Apart from flow control, TCP provides the retransmission mechanism. The receiver needs to acknowledge the receiving of data sent from the sender. If the sender fails to receive the acknowledgement from the receiver, the sender considers that the data is lost and retransmits the lost data.

With such flow control and retransmission mechanisms, TCP can suppress traffic burst and prevent packet loss. Stuttering will not occur if TCP is adopted. This is why the video effect delivered by TCP is better than that delivered by UDP in actual environments.

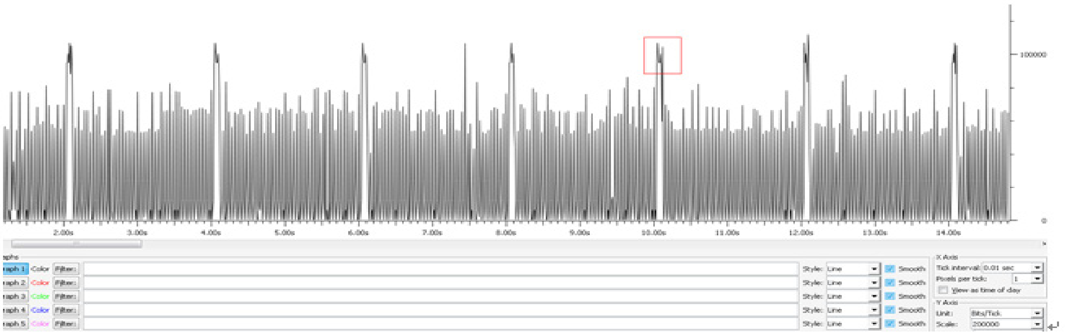
• UDP-based Flow Control Test

To verify the burst traffic suppression effect offered by flow control, one flow control mechanism is simulated on the basis of UDP.

The traffic burst comes from IPCs. Therefore, run the speed command on the port that connects an IPC to a switch, to forcibly negotiate the rate to 10 Mbps. Configure CBR and UDP, set the bit rate to 2 Mbps, resolution to 720P, frame rate to 25 fps, and GOP to 50 for the IPC. The 10Mbps rate is sufficient for the average traffic (2 Mbps) of one video stream. The port rate is restricted to 10 Mbps, and therefore, the maximum burst traffic sent from the IPC to the switch can only be 10 Mbps, achieving burst traffic smoothing and shaping.

Use the same settings used in traffic characteristics analysis to capture waveforms. The figure below shows a waveform of a static picture.

Figure 30

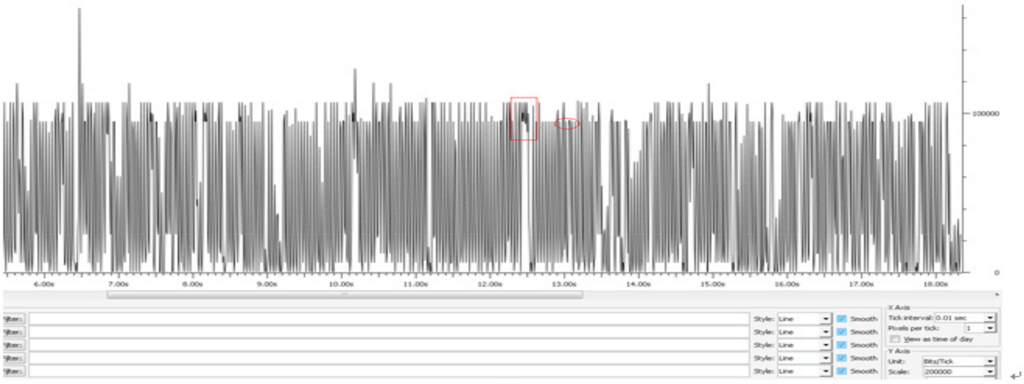


Style: Line
Tick interval: 0.01 sec
Pixels per tick: 1
Unit: Bits/Tick
Scale: 200000

Obviously, the I frame pattern (marked in a red rectangle) changes greatly. The burr height is restricted at about 100 kbit at the sampling frequency of 10 ms, that is, 100 Mbit/s. In addition, the width of I frame burrs is larger, indicating that the IPC takes a longer time to send I frames. The two changes show that the rate-limit-based flow control is effective and implements traffic smoothing and shaping.

The figure below shows the waveform of a motion picture.

Figure 31



Style: Line
Tick interval: 0.01 sec
Pixels per tick: 1
Unit: Bits/Tick
Scale: 200000

The traffic of P frames becomes heavier but is restricted within 100 kbit. I frames and P frames are distinguished by burr width instead of height. Burrs with a larger width indicate I frames (marked in a red rectangle) while burrs with a smaller width indicate P frames (marked in a red circle).

It is observed that the video pictures are played very smoothly without stuttering. The IPC can send packets at the rate of the 10 Mbps at most due to rate limit. Packets that cannot be sent out temporarily are buffered in the IPC. The IPC has a buffer large enough to buffer I frames and the first P frames.

Connect eight IPCs and continue to perform the test. The egress bandwidth can be restricted to 20 Mbps when no rate limit is configured, and can be restricted to about 18.5 Mbps after rate limit is configured, and the buffer requirement is significantly lowered.

The test shows that UDP can ensure no packet loss after flow control is applied. Owing to the flow control and retransmission mechanisms, the video effect delivered by TCP is superior to that delivered by UDP.

• Superimposition of Preview Streams and Storage Streams

The preceding sections analyze the scenario in which multiple video streams are superimposed on the switch.

There is a typical application scenario that needs to be analyzed: One IPC outputs two video streams: one preview stream that is output to display devices such as the TV wall, and one storage stream that is output to storage devices such as the Network Video Repository (NVR). The storage stream must be TCP stream based on the existing IPC and NVR. The preview stream can be TCP or UDP stream.

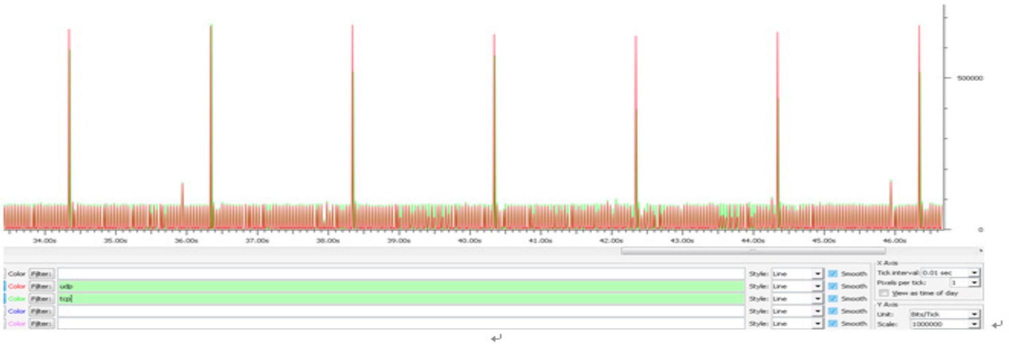
In the mainstream applications of IP surveillance in current intelligent buildings, the bit rate is set to 2 Mbps for both the preview stream and storage stream, totaling 4 Mbps.

Configure CBR, and set the resolution to 720P, frame rate to 25 fps, and GOP to 50.

When two TCP streams are superimposed, no packet loss will occur due to the flow control and retransmission mechanisms of TCP.

The superimposition of a UDP stream and a TCP stream is analyzed as follows: The figure below shows a captured waveform of the superimposed UDP stream and TCP stream.

Figure 32



Filter: udp — Style: Line — Smooth
Filter: tcp — Style: Line — Smooth
Tick interval: 0.01 sec
Pixels per tick: 1
Unit: Bits/Tick
Scale: 1000000

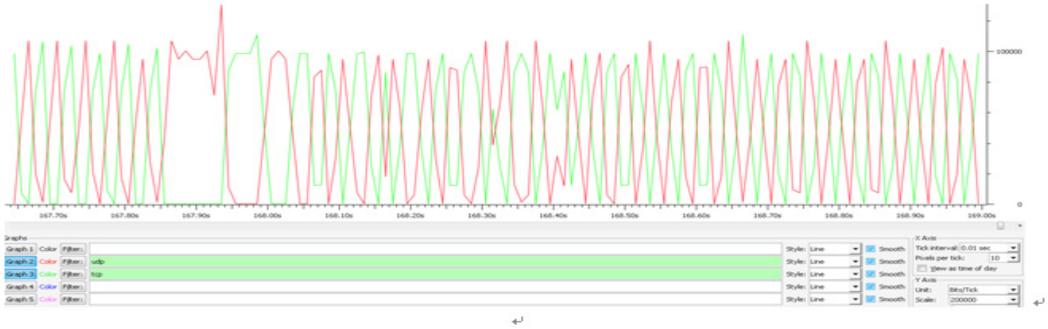
The red section indicates the UDP stream while the green section indicates the TCP stream.

The two video streams present a similar burst form. Even if the frame rate and I frame interval are set to different values for the two streams, the video streams actually sent by the IPC still have the same frame rate and I frame interval. The IPC selects one configuration (the later configuration takes effect) from the configurations for the TCP stream and UDP stream, to apply to both video streams.

The two video streams are sampled at the same moment. Therefore, the I frames and P frames of the two video streams are superimposed separately. Especially, the superimposition of I frames doubles the burst traffic.

Use the rate limit method again to simulate flow control. The figure below shows captured waveforms.

Figure 33



Graph 1 Color Filter: Style: Line ☑ Smooth
Graph 2 Color Filter: udp Style: Line ☑ Smooth
Graph 3 Color Filter: tcp Style: Line ☑ Smooth
Graph 4 Color Filter: Style: Line ☑ Smooth
Graph 5 Color Filter: Style: Line ☑ Smooth

X Axis
Tick interval: 0.01 sec
Pixels per tick: 10
□ View as time of day

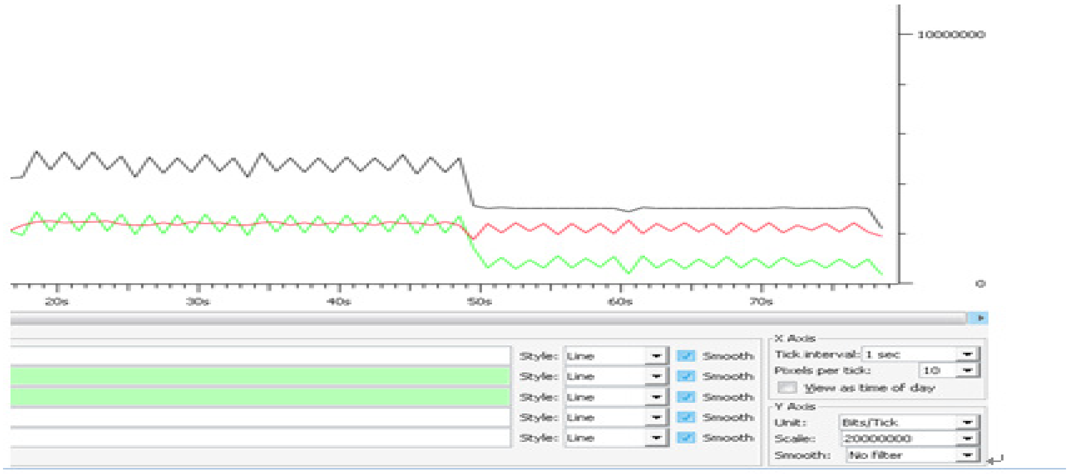
Y Axis
Unit: Bits/Tick
Scale: 200000

The 10 Mbit/s port bandwidth restricts both the UDP traffic and TCP traffic, and this enables alternate distribution of the burrs of the two video streams, achieving an excellent flow control effect for the two superimposed streams. The actual video effects of both pictures are smooth and no exception occurs.

It is found from further packet capture that the TCP receive window is set to 32768, indicating that the IPC can send a maximum of 32768 bytes instantaneously via TCP. Owing to this receive window, the rate limit method also smooth the TCP stream on which flow control is performed.

In addition, the superimposition of the UDP stream and TCP stream is analyzed in the case of small egress bandwidth. TCP provides the flow control mechanism while UDP sends packets in best-effort manner. After rate limit is applied to TCP packets, UDP packets consume most bandwidth. The TCP receive window is always fully occupied, and the TCP traffic is even decreased. The figure below shows a captured waveform.

Figure 34



Style: Line ☑ Smooth
Style: Line ☑ Smooth
Style: Line ☑ Smooth
Style: Line ☑ Smooth
Style: Line ☑ Smooth

X Axis
Tick interval: 1 sec
Pixels per tick: 10
□ View as time of day

Y Axis
Unit: Bits/Tick
Scale: 200000000
Smooth: No filter

In summary, traffic burst is a basic characteristic in current video networks, and burst traffic superimposition is ubiquitous in various application scenarios. The traffic burst capacity of IPCs, especially the burst traffic superimposition, goes far beyond the buffer capacity of the switch, which leads to packet loss and even stuttering. TCP provides the flow control and retransmission mechanisms, and is capable of smoothing burst traffic and preventing packet loss. TCP can deliver a smoother picture effect than UDP.

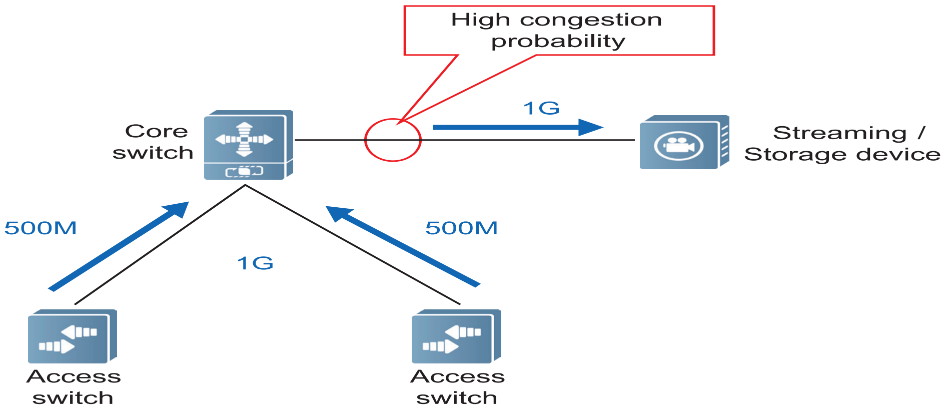
Ruijie Video Transmission Optimization Solution

• Video Traffic Smoothing and Shaping

As described in previous chapters, video traffic is characterized by strong burst. When burst traffic of multiple video streams is superimposed, the traffic may easily exceed the reserved port bandwidth and even the buffer capacity of the switch, leading to packet loss and ultimately affecting the video play effect.

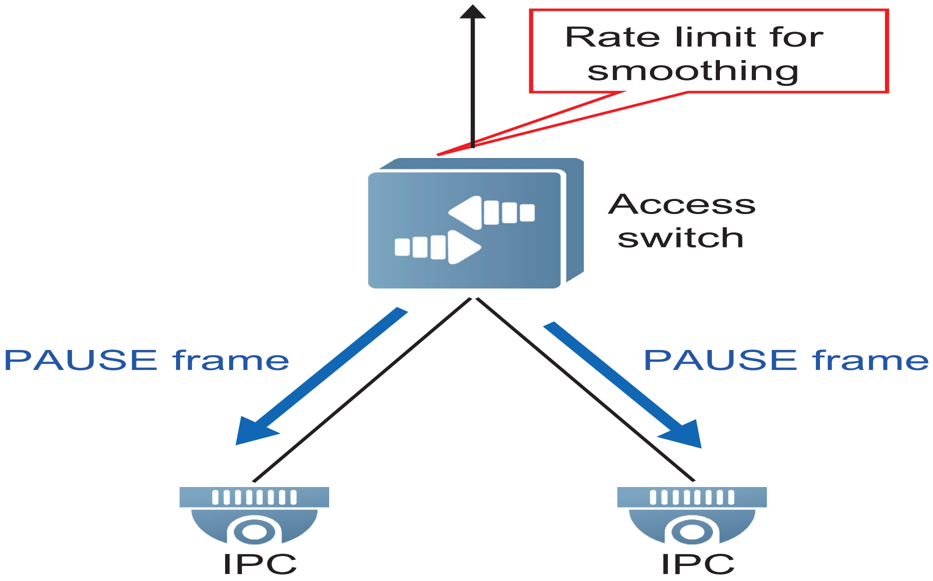
For this, Ruijie puts forward the flow control backpressure solution, which enables the switch S26I to associate with IPCs to press burst traffic beyond the buffer capacity of the switch back to the IPCs for buffering, thereby smoothing traffic burst. In this way, uplink traffic can be output at a low rate steadily, without affecting the video preview quality.

Figure 35



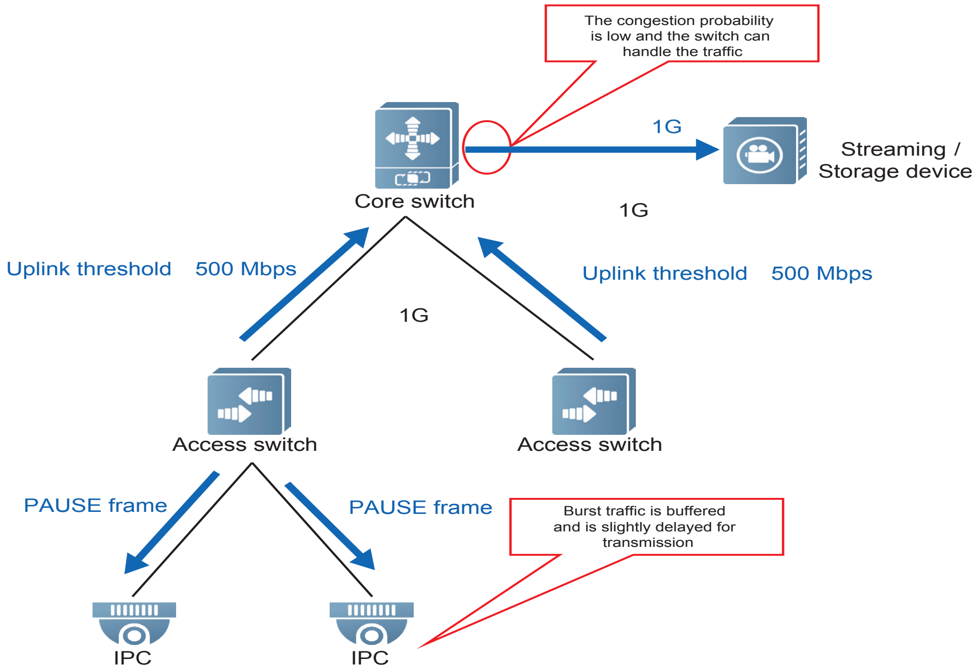
As shown in Figure 34, assume that the average traffic of uplink ports of the two access switches is 500 Mbps each, totaling 1 Gbps. The total traffic exceeds 1 Gbps bandwidth in the case of traffic burst. As a result, the uplink port of the core switch connected to a storage device is congested, causing packet loss. Delay, stuttering, and pixilation will occur during video play.

Figure 36



As shown in Figure 35, the flow control backpressure solution is applied to the access switch to suppress traffic burst from the source, that is, the flow control function is enabled on the port connected to the downlink IPCs (IPCs need to support flow control), and rate limit is applied to the port connected to the uplink aggregation switch for smoothing, so that uplink traffic is controlled within a low and stable threshold, thereby alleviating the bandwidth pressure of the uplink device. The figure below shows the final result.

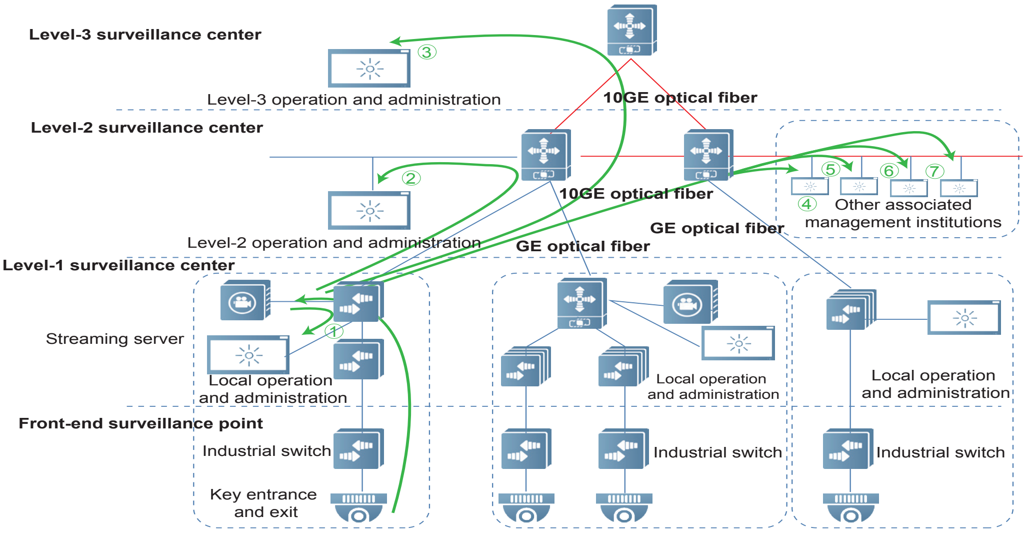
Figure 37


The congestion probability is low and the switch can handle the traffic
1G
Core switch
Streaming / Storage device
1G
Uplink threshold 500 Mbps
Uplink threshold 500 Mbps
1G
Access switch
Access switch
PAUSE frame
PAUSE frame
Burst traffic is buffered and is slightly delayed for transmission
IPC
IPC

- Multicast Application

There are usually multiple on-demand hotspot IPCs and each IPC sends one unicast stream, which is duplicated by the streaming server into multiple streams. In this case, the streaming server may experience a range of problems such as processing bottleneck, bandwidth bottleneck, delay, Single Point of Failure (SPOF), high cost, and complex network architecture.

Figure 38 Solution Dependent on Duplication and Distribution of the Streaming Server


Level-3 surveillance center
③
Level-3 operation and administration
10GE optical fiber
Level-2 surveillance center
②
⑤
⑥
⑦
④
Other associated management institutions
10GE optical fiber
GE optical fiber
Level-2 operation and administration
GE optical fiber
Level-1 surveillance center
Streaming server
①
Local operation and administration
Local operation and administration
Local operation and administration
Front-end surveillance point
Industrial switch
Industrial switch
Industrial switch
Key entrance and exit

For this, Ruijie proposes to use the mature multicast network technology to transmit audio and video data streams, which has the following advantages:

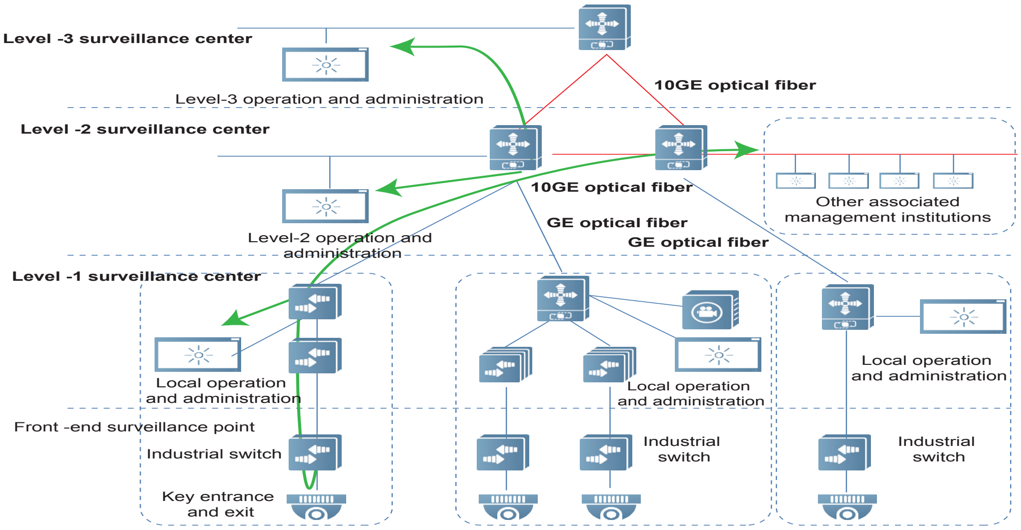
* **The streaming server is not required, which reduces the building cost and simplifies the network architecture.**

* **Only one copy of data is transmitted through the network channels, which saves the network bandwidth.**

* **Multicast data is transmitted via UDP packets. The flow control backpressure technology described in "Video Traffic Smoothing and Shaping" eliminates the UDP unreliability problem, and delivers a video transmission effect not second to TCP.**

* **The bridging technology is adopted to prevent broadcast storms that may be brought by front-end devices.**

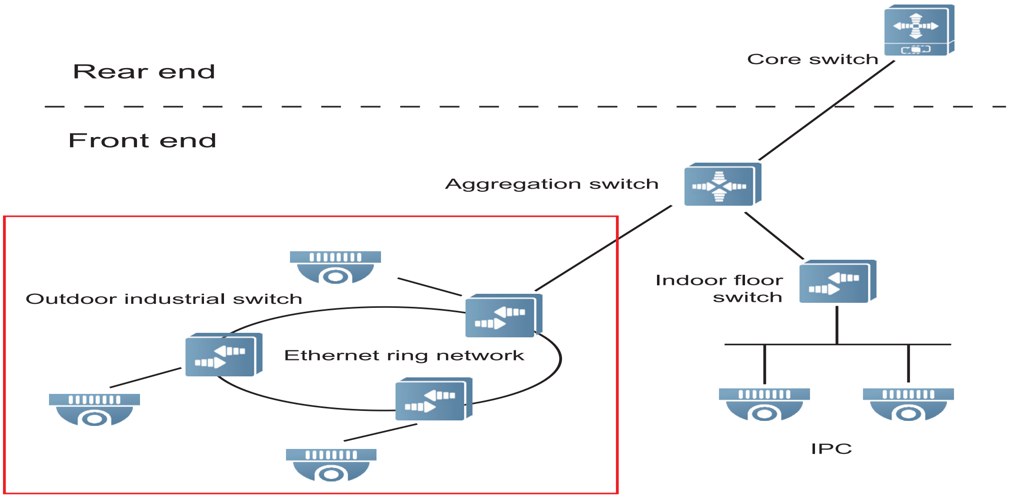
Figure 39 Multicast Application Solution


Level -3 surveillance center
Level-3 operation and administration
10GE optical fiber
Level -2 surveillance center
10GE optical fiber
Other associated management institutions
GE optical fiber
GE optical fiber
Level-2 operation and administration
Level -1 surveillance center
Local operation and administration
Local operation and administration
Local operation and administration
Front -end surveillance point
Industrial switch
Industrial switch
Industrial switch
Key entrance and exit

• ERPS

The Ethernet Ring Protection Switching (ERPS) protocol, also called G.8.232, is a link layer protocol developed by the ITU for Ethernet ring networks. It can prevent broadcast storms caused by data loops when an Ethernet ring network is intact, and can rapidly recover the communication between nodes on the ring network when a link on the network is disconnected.

Figure 40 Multicast Application Solution



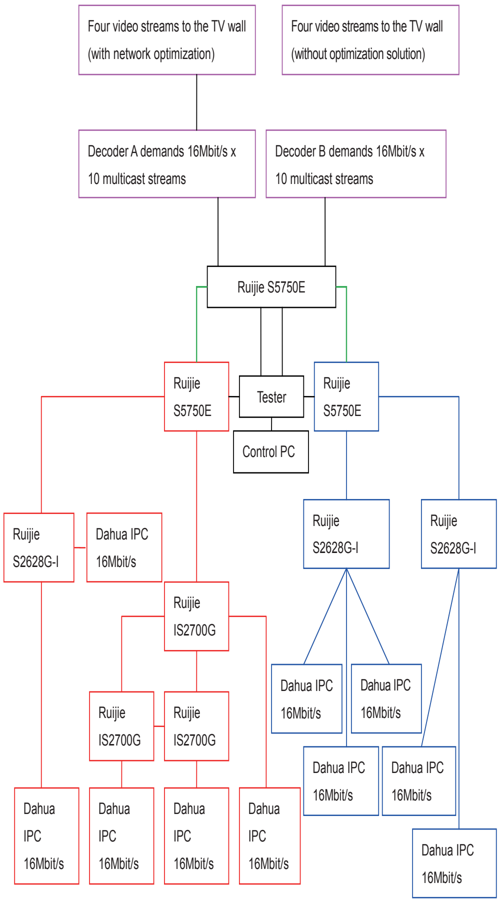
As shown in Figure 39, in extreme outdoor environments, the ring network topology can be used to improve the network reliability. When any node on the ring network malfunctions, the network communication can be recovered rapidly, without affecting services.

• Solution Demonstration in Wuhan Intelligent Building Annual Meeting 2014

Figure 41 shows the network topology presented to customers at the site of Wuhan intelligent building annual meeting. The section marked in red frames indicate a network in which the Ruijie video transmission optimization solution is applied. In this section, flow control is enabled on the S2628G-I and three IS2700G devices, rate limit is configured on the port connected to the uplink S5750E for traffic shaping, and the three IS2700G devices compose an ERPS ring network. The section marked in blue frames indicates a common network without any optimization configuration. Both decoders demand video streams of all IPCs in the network (that is, the video stream of each IPC is demanded by two decoders simultaneously). With the multicast technology, each IPC outputs only one data stream, alleviating the network load.

To emphasize the alleviation on bandwidth pressure of uplinks (links marked in green in the figure) by the optimization solution, Ruijie uses a tester to retain certain traffic load in uplinks, which occupy 85-90% of the total bandwidth. Only remaining bandwidth is used to transmit video data streams. Decoder A and Decoder B access the TV wall to respectively preview the video play effect of the four video streams from IPCs (optimized) in the network marked in red frames, and from IPCs (not optimized) in the network marked in blue frames. It can be obviously seen that when network bandwidth is limited, obvious stuttering occurs in the preview of the four video streams from the network marked in blue frames, while the video preview of the four video streams from the network marked in red frames is still smooth.

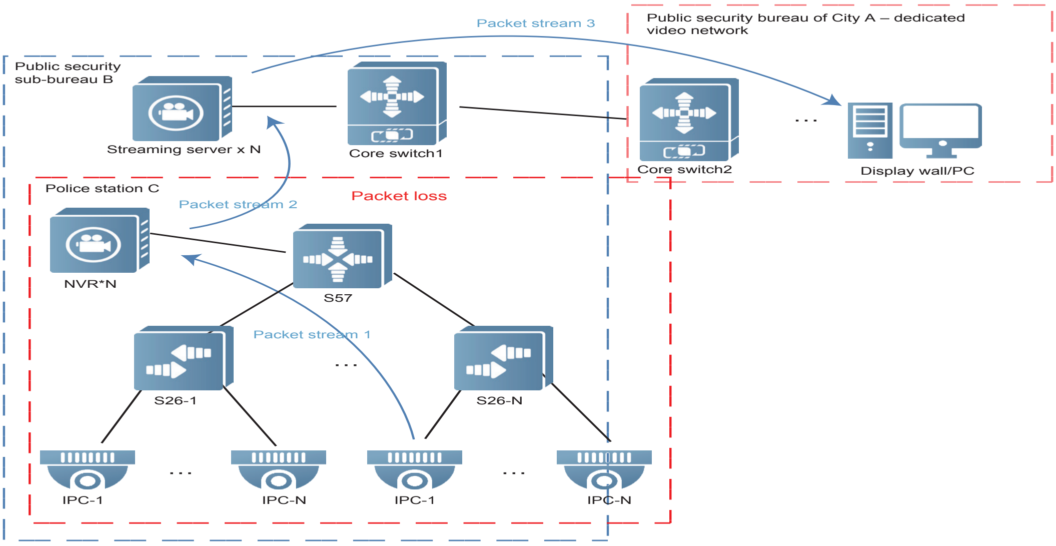
Figure 41



Stuttering Cases

• Stuttering Case of a Safe City

Figure 42



The figure above shows the typical topology of the video dedicated network of a safe city. The topology covers three layers: police station, district public security sub-bureau, and municipal public security bureau. In the video dedicated network of the public security bureau of City A, when video are previewed on a client, the video packet streams are packet streams 1–3 in the figure. When videos from IPCs in Police Station C of District B are previewed, obvious stuttering exists. The traffic statistics of ports of all switches are checked based on the direction of packet streams. It is found that the 1G uplink port of the S5750E that serves as an aggregation device encounters continuous packet loss. In addition, when some other S5750Es of the police station are checked, it is found that their 1G uplink ports generally experience continuous packet loss, and packet loss occurs when the uplink traffic reaches about 100 Mbps.

Packet loss occurring on the uplink ports of the S5750E is relevant to the buffer allocation. The safe city video dedicated network of City A uses S5750E 10.4(3b16). A single 1G port can consume a maximum of 300 buffer units (each buffer unit can store 256 bytes of data), which are insufficient for the video traffic burst in the network. Actually, an S5750E has 6000 buffer units, and considerable buffer resources are wasted in this scenario.

Use the preview of video streams from IPC D in Police Station C to perform the following contrast test:

1.Adjust the buffer (the uplink traffic of the S5750E is 160 Mbps in this test).

* **Before the buffer is adjusted, observe the video preview twice for 5 minutes each time. It is found that stuttering occurs for 16 and 17 times respectively, and 6309 and 7337 packets are lost on the uplink port respectively.**

* **Adjust the buffer unit consumption limit to 1200. Observe video preview twice for 5 minutes each time. It is found that stuttering occurs for 7 and 6 times respectively, and no packet loss occurs on the uplink port.**

2.Replace the streaming server and adjust the buffer (the uplink traffic of the S5750E is about 140 Mbps in this test).

* **Before the buffer is adjusted, observe the video preview three times for 2 minutes each time. It is found that stuttering occurs twice, twice, and once respectively.**

* **Adjust the buffer unit consumption limit to 1200. Stuttering occurs occasionally and no stuttering occurs within the 2 minutes.**

Use the same method to test the video preview of other IPCs. The improvement effect is unstable and is not as outstanding as that in the case above. It can be determined that severe packet loss occurring on the uplink ports of the S5750Es is an factor that causes stuttering. The transmission optimization solution described in 5 "Ruijie Video Transmission Optimization Solution" can effectively alleviate the load of the uplink ports.

Appendix

• Analysis of Video Impact Factors

Video Impact Factors

Factors that affect the video quality are as follows:

* **Quality of video data that is output by the video headend: Test the quality of Moving Picture Experts Group (MPEG)-2 Transport Stream (TS) video streams in accordance with TR101-290, or test the quality of video streams in accordance with ITU BT.500.**

* **Capability of STAs in receiving and processing video data**

* **Impact of the bearer network on video data**

1. Delay: delay in the time of watching videos. The video play delay does not affect the video preview quality.
2. Jitter: Network congestion and changes in network device performance will cause jitter in video streams. Monitoring video stream jitter can help O&M personnel identify the trend of video transmission quality deterioration in advance.
3. Packet loss: Packet loss directly affects the video play quality. The loss of any type of packets (I frames or P frames) will lower the video play quality to different extents.

Media Delivery Index (MDI) detection is used to monitor the quality of the bearer network, and involves the following factors:

* **Delay Factor (DF)**

DF indicates the delay and jitter statuses of video streams, in the unit of ms. .A video decoding device uses a video buffer to shield the impact of network delay and jitter on the video play quality. The video decoding device can determines the size of the buffer based on the DF value. The DF value detected by the bearer network device and preset size of the STA buffer can be used in combination, to judge whether network delay and jitter affect the video play quality of STAs.

Higher jitter indicates a greater DF value. When the duration of video content accommodated in the buffer is not smaller than the DF value of tested video streams, the video play quality will not deteriorate.

* **Theoretical DF value**

A theoretical DF value refers to the DF value obtained when media streams are transmitted on a network without congestion and no delay and jitter occur. It is calculated as follows:

DF = Size of (IP)/MR

Size of (IP) indicates the size of an IP packet. For example, each IP packet contains seven MPEG-2 TS packets and the size of each MPEG-2 TS packet is 188 bytes. The value of Size of (IP) is 1316 bytes (that is, 7 x 188 bytes). MR indicates the rate of data streams, in the unit of bytes/second.

* **Actual DF value**

An actual DF value is the DF value obtained when media streams are transmitted on the actual network. It is calculated as follows:

The virtual buffer size of a test point is VB.

VB(i,pre) = Sum (Sj) – MR * Ti; where j = 1..i–1

VB(i,post) = VB(i,pre) + Si

VB(i,pre) indicates the VB value before the ith data packet is received.

VB(i,post) indicates the VB value after the ith data packet is received.

Sj indicates the size of the jth data packet.

Ti indicates the arrival time of the ith data packet (in relative to the test start time).

MR indicates the rate of data streams, in the unit of bytes/second.

If k packets are received within a detection period, there are a total of 2*k+1 VBs.

VB(max) indicates the maximum value among the 2*k+1 VBs in the detection period.

VB(min) indicates the minimum value among the 2*k+1 VBs in the detection period.

DF = [VB(max) – VB(min)]/ MR

* **Media Loss Rate (MLR)**

The unit of MLR is the number of media packets lost per second. MLR indicates the transmission packet loss rate of video streams. The loss of video information packets will directly affect the video play quality, and ideal IP video stream transmission requires the MLR value to be zero.

The formula for calculating the MLR is as follows:

MLR = Number of lost media packets/Sampling period

Media packets refer to valid MPEG-2 TS packets excluding filled packets. If an IP packet contains seven MPEG-2 TS packets and the seven packets do not contain filled packets, the loss of an IP packet is considered as the loss of seven media packets.

Note: MLR indicates the packet loss rate every second, MLR-15 indicates the packet loss rate every 15 minutes, and MLR-24 indicates the packet loss rate every 24 hours.

Recommended MDI Thresholds

The MDI thresholds are recommended as follows:

* **DF = 50 ms**

* **MLR = 8 media packets/second**

* **MLR-15 = 128 media packets/second**

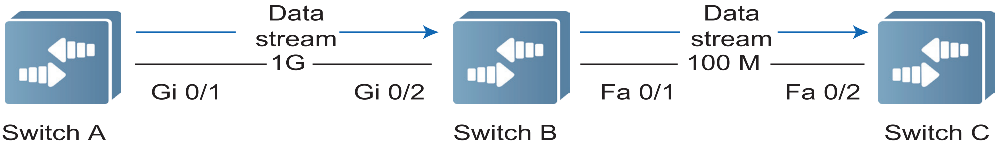
* **MLR-24 = 1024 media packets/second**

• Flow Control

Work Principle

According to the PAUSE mechanism defined in the IEEE 802.3, when a receiver is incapable of processing received packets, the receiver needs to notify the sender to temporarily stop sending packets, to prevent packet loss. This process is implemented by receiving and sending common PAUSE frames over ports.

Figure 43



As shown in the figure above, when Port Gi 0/1 and Port Gi 0/2 forward packets at a rate of 1 Gbps, Port Fa 0/1 will be congested. Enable the common flow control function on Port Gi 0/1 and Port Gi 0/2, to prevent packet loss:

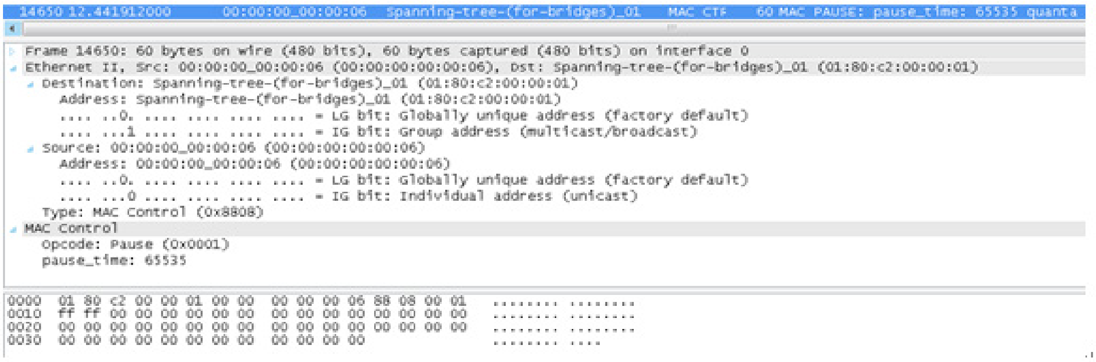
* **When Port Fa 0/1 is congested during packet forwarding, Switch B buffers the packets. When the traffic exceeds the forwarding capacity of Switch B, packet loss occurs. In this case, Port Gi 0/2 sends a common PAUSE frame to Port Gi 0/1, so that Port Gi 0/1 is notified to temporarily stop sending packets.**

* **After receiving the PAUSE frame, Port Gi 0/1 temporarily stops sending packets to Port Gi 0/2. The pause time is carried in the PAUSE frame. When congestion persists, the switch repeats the preceding steps to conduct step-by-step backpressure operation until congestion is eliminated.**

Although the PAUSE mechanism can prevent packet discarding but will block all traffic on a link. In essence, it temporarily stops the entire link.

PAUSE Packet Format

Figure 44



```
14650 12.441912000      00:00:00_00:00:06   Spanning-tree-(for-bridges)_01    MAC CTF    60 MAC PAUSE: pause_time: 65535 quanta

Frame 14650: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
Ethernet II, Src: 00:00:00_00:00:06 (00:00:00:00:00:06), Dst: Spanning-tree-(for-bridges)_01 (01:80:c2:00:00:01)
  Destination: Spanning-tree-(for-bridges)_01 (01:80:c2:00:00:01)
    Address: Spanning-tree-(for-bridges)_01 (01:80:c2:00:00:01)
    .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
    .... ...1 .... .... .... .... = IG bit: Group address (multicast/broadcast)
  Source: 00:00:00_00:00:06 (00:00:00:00:00:06)
    Address: 00:00:00_00:00:06 (00:00:00:00:00:06)
    .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
    .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
  Type: MAC Control (0x8808)
MAC Control
  Opcode: Pause (0x0001)
  pause_time: 65535

0000  01 80 c2 00 00 01 00 00  00 00 00 06 88 08 00 01   ........ ........
0010  ff ff 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ........ ........
0020  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ........ ........
0030  00 00 00 00 00 00 00 00  00 00 00 00              ........ ....
```

The figure above shows a PAUSE frame, in which the destination MAC address is fixed to 01-80-C2-00-00-01, Ethertype is fixed to 0x8808, Opcode (operation code) is fixed to 0x0001, pause_time occupies two bytes and indicates the required pause duration for the peer end, and the unit is the time required to transmit 512 bits at the current transmission rate.

Ruijie Networks Co.,Ltd

For further information,please visit our website **http://www.ruijienetworks.com**