

版权

1 订阅 16 篇文章 订阅专栏

在本文中，将介绍h264通过nrtplib库来现实时的发送和接收。发送方发送，按照编码的习惯，我们习惯上是每完成一帧数据的编码就整一帧数据的发送。与此类似，在接收端，我们也喜欢将一整帧的数据接收完整后再进行解码等处理。但是，这样会带来一个问题，我们网络最大传输单位MTU一般是1400个字节，而我们进行h.264数据编码的时候，很多数据帧会大于1400帧，所以需要分开几个数据包来发送，在接收的时候，需要将开包的数据包再合成一个数据帧。

下面代码是在我电脑写的最简程序。

发送端IP: 192.168.0.5

接收端IP: 192.168.0.6

发送端口: 6666

接收端口: 6664

注意几点:

- (1) jrtplib不能使用基端口，基端口用来建立RTCP
- (2) jrtplib设置的默认MUT是1400，但是实际每包数据不能发送1400字节，因为它还有TCP 头数据

receive.c

```

1  /*
2      *   FileName: receive.cpp
3      *   Desc: receive h.264 from RTP server
4      *   Author: licaibiao
5      *   LastChange: 2017-04-22
6      *   =====
7  #include <jrtplib3/rtpsession.h>
8  #include <jrtplib3/rtpudpv4transmitter.h>
9  #include <jrtplib3/rtpipw4address.h>
10 #include <jrtplib3/rtpsessionparams.h>
11 #include <jrtplib3/rtperrors.h>
12 #include <jrtplib3/rtplibraryversion.h>
13 #include <jrtplib3/rtppacket.h>
14 #include <stdlib.h>
15 #include <stdio.h>
16 #include <iostream>
17 #include <string>
18
19 using namespace jrtplib;
20
21 void checkerror(int rtperr)
22 {
23     if (rtperr < 0)
24     {
25         std::cout << "ERROR: " << RTPGetErrorString(rtperr) << std::endl;
26         exit(-1);
27     }
28 }
29
30 int main(void)
31 {
32     RTPSession sess;
33     uint16_t portbase = 6664;
34     int status;
35     bool done = false;
36
37     RTPUDPV4TransmissionParams transparams;
38     RTPSessionParams sessparams;
39     sessparams.SetOwnTimestampUnit(1.0/10.0);
40
41     sessparams.SetAcceptOwnPackets(true);
42
43     transparams.SetPortbase(portbase);
44     status = sess.Create(sessparams,&transparams);
45     checkerror(status);
46
47     sess.BeginDataAccess();
48     RTPTime delay(0.001);
49     RTPTime starttime = RTPTime::CurrentTime();
50
51     FILE *fd;
52     size_t len;
53     uint8_t *loaddata;
54     RTPPacket *pack;
55     uint8_t buff[1024*100] = {0};
56     int pos = 0;
57
58     fd = fopen("./test_recv.h264","wb+");
59     while (!done)
60     {
61         status = sess.Poll();
62         checkerror(status);
63
64         if (sess.GotoFirstSourceWithData())
65         {
66             do
67             {
68                 while ((pack = sess.GetNextPacket()) != NULL)
69                 {
70
71                     loaddata = pack->GetPayloadData();
72                     len = pack->GetPayloadLength();
73
74                     if(pack->GetPayloadType() == 96) //H264
75                     {
76                         if(pack->HasMarker()) // the last packet
77                         {
78                             memcpy(&buff[pos],loaddata,len);
79                             fwrite(buff, 1, pos+len, fd);
80                             pos = 0;
81                         }
82                         else
83                         {
84                             memcpy(&buff[pos],loaddata,len);
85                             pos = pos + len;
86                         }
87                     }
88                 }
89             } while (1);
90         }
91     }
92 }
93 */

```

```

87         }else
88         {
89             printf("!!! GetPayloadType = %d !!! \n ",pack->GetPayloadType());
90         }
91
92         sess.DeletePacket(pack);
93     }
94     } while (sess.GotoNextSourceWithData());
95 }
96
97 RTPTime::Wait(delay);
98 RTPTime t = RTPTime::CurrentTime();
99 t -= starttime;
100 if (t > RTPTime(40.0))
101     done = true;
102 }
103 fclose(fd);
104
105 sess.EndDataAccess();
106 delay = RTPTime(10.0);
107 sess.BYEDestroy(delay,0,0);
108
109 return 0;
110 }

```

发送端程序 sender

```

1  /*=====
2  *   FileName: sender.cpp
3  *   Desc: sending h.264 data to client
4  *   Author: licaibiao
5  *   LastChange: 2017-04-22
6  *   =====*/
7  #include <jrtpplib3/rtpsession.h>
8  #include <jrtpplib3/rtpudpv4transmitter.h>
9  #include <jrtpplib3/rtpipv4address.h>
10 #include <jrtpplib3/rtpsessionparams.h>
11 #include <jrtpplib3/rtperrors.h>
12 #include <jrtpplib3/rtplibraryversion.h>
13 #include <stdlib.h>
14 #include <stdio.h>
15 #include <iostream>
16 #include <string>
17
18 using namespace jrtpplib;
19
20 #define MAXLEN    (RTP_DEFAULTPACKETSIZE - 100)
21
22 void checkerror(int rtperr){
23     if (rtperr < 0){
24         std::cout << "ERROR: " << RTPGetErrorString(rtperr) << std::endl;
25         exit(-1);
26     }
27 }
28
29 class MyRTPSession : public RTPSession{
30 public:
31     MyRTPSession(void);
32     ~MyRTPSession(void);
33     void SendH264Nalu(RTPSession* sess,uint8_t* m_h264Buf,int buflen);
34 protected:
35
36
37 };
38
39 MyRTPSession::MyRTPSession(void){}
40 MyRTPSession::~MyRTPSession(void){}
41
42 int main(void)
43 {
44     int i;
45     int num;
46     int status;
47
48     RTPSession sess;
49     MyRTPSession sender;
50     uint16_t portbase = 6666;
51     uint16_t destport = 6664;
52     uint8_t destip[]={192,168,0,6};
53
54     RTPUDPV4TransmissionParams transparams;
55     RTPSessionParams sessparams;
56
57     /* set h264 param */
58     sessparams.SetUsePredefinedSSRC(true); //设置使用预先定义的SSRC
59     sessparams.SetOwnTimestampUnit(1.0/9000.0); /* 设置采样间隔 */
60     sessparams.SetAcceptOwnPackets(true); //接收自己发送的数据包
61
62     transparams.SetPortbase(portbase);
63     status = sess.Create(sessparams,&transparams);
64     checkerror(status);
65
66     RTPIPv4Address addr(destip,destport);
67     status = sess.AddDestination(addr);
68     checkerror(status);
69
70     sess.SetDefaultTimestampIncrement(3600);/* 设置时间戳增加间隔 */
71     sess.SetDefaultPayloadType(96);
72     sess.SetDefaultMark(true);
73
74     FILE *fd;
75     int pos = 0;
76     int header_flag = 0;
77     uint8_t buff[1024*100] = {0};
78
79     fd = fopen("./test.h264","rb");
80     fread(buff, 1, 4, fd);
81     if((buff[0]==0)&&(buff[1]==0)&&(buff[2]==0)&&(buff[3]==1)){
82         header_flag = 1;
83         pos = 4;
84     }else{
85         header_flag = 0;
86         pos = 3;
87     }
88
89     while((feof(fd)==0))
90     {
91         buff[pos++] = fgetc(fd);
92
93         if(header_flag == 1){ //00 00 00 01
94             if((buff[pos-1]==1)&&(buff[pos-2]==0)&&(buff[pos-3]==0)&&(buff[pos-4]==0)){

```

```
95         sender.SendH264Nalu(&sess, buff, pos-4);
96         buff[0] = 0x00;
97         buff[1] = 0x00;
98         buff[2] = 0x00;
99         buff[3] = 0x01;
100         pos = 4;
101
102         RTPTime::Wait(0.03);
103     }
104
105     }
106     else{
107         if((buff[pos-1]==1)&&(buff[pos-2]==0)&&(buff[pos-3]==0)){
108             sender.SendH264Nalu(&sess, buff, pos-3);
109             buff[0] = 0x00;
110             buff[1] = 0x00;
111             buff[3] = 0x01;
112             pos = 3;
113
114             RTPTime::Wait(0.03);
115         }
116     }
117
118     }
119     if(pos != 0){
120         sender.SendH264Nalu(&sess, buff, pos);
121     }
122     fclose(fd);
123     printf("end of the read\n");
124     sess.BYEDestroy(RTPTime(10,0),0,0);
125     return 0;
126 }
```

在上面的代码中，去掉了void SendH264Nalu(RTPSession* sess,uint8_t* m_h264Buf,int buflen) 函数的实现

原文链接： https://blog.csdn.net/li_wen01/article/details/70435005

🔗 文章知识点与官方知识档案匹配，可进一步学习相关知识

CS入门技能树 > Linux入门 > 初识Linux 34714 人正在系统学习中

基于JRTPLIB库的H_264视频传输系统.pdf	03-23
基于JRTPLIB库的H264视频传输系统，利用JRTPLIB库传输H264视频，包括分包传输控制方案	
使用jrtpIib发送和接收网络流（支持解码、播放） 基于jrtpIib实现了RTP发送和接收功能，支持发送H264、PS、TS等文件流，接收端读取发送端发来的数据后，取Payload数据，通过FFmpeg分离、解码...	01-15
使用JrtpIib(RTP)传输H.264视频文件(1) 文件传输统计zero bit_TooyLee... H.264Payload 格式定义了三种不同的基本的负载(Payload)结构。接收端可能通过RTP Payload的第一个字节来识别它们。这一个字节类似NALU头的格式。...	8-18
使用JrtpIib(RTP)传输H.264视频文件_wh8_2011的博客 J RTP传输: 好了,下面是我写的使用J RTP进行发送H264数据包的例子.具体解释可以看注释。发送端也可以接收接收端发送过来的RTCP数据包。#define ...	9-3
Linux下JRTPLIB的编译安装 (76条消息) linux下编译jrtpIib-3.9.1_jrtp Iib编译_草上爬的博客-CSDN博客	qq_42475191的博客 06
读取H264+发送RTP包的例子代码-VsParserSendH264 读取H264文件，获得每一帧的数据，然后切分成NAL，封装成RTP包，用UDP发送到网络。	02-02
毕业设计之JrtpLib H264(裸视频数据)实时视频传输(发送与接受)_lhy... y_jrtp.cpp #include "y_jrtp.h"staticboolCheckError(inttperr) {if(tperr <0) {std::cout<<"ERROR: "<<"RTPGetErrorString(tperr)"<<std::endl;returnfalse; }r...	8-25
...传输H.264视频文件(2)_jrtpIib 发送 h264_ htx99的博客 voidCRTPSender::SendH264Nalu(unsignedchar* m_h264Buf,intbuflen) { unsignedchar*pSendbuf;//发送数据指针 pSendbuf = m_h264Buf; //去除前导码...	8-16
利用JrtpIib库发送h264文件-vc播放 本资源给出了用jrtpIib发送h264文件的流程，可供参考	07-14
使用RTP协议发送H264数据包 使用RTP协议发送H264数据包。通过设置目的IP地址和端口号，使用VLC播放器解码播放H264视频。源码有详细的解释。	04-19
使用JrtpIib(RTP)传输H.264视频文件(2)_str999_cn的博客 J RTP传输: 好了,下面是我写的使用J RTP进行发送H264数据包的例子.具体解释可以看注释。发送端也可以接收接收端发送过来的RTCP数据包。#define ...	8-4
使用JrtpIib传输H.264视频文件(3)_TooYLee的博客 使用JrtpIib传输H.264视频文件(3) 转自http://blog.csdn.net/liushu1231/article/details/9203643 介绍如何发送h264,下面我们介绍下如何接收h264文件。其...	8-17
FFmpeg4.3系列之24: RTP及jrtpIib发送H.264码流 敬告: 该系列的课程在抓紧录制更新中, 敬请大家关注。敬告: 该系列的课程涉及: FFmpeg,WebRTC,SRS,Nginx,Darwin,Live555,OpenCV等。包括: ...	09-01
JrtpIib学习目录及总结 RTP(实时传输协议)一般比较少单独的使用，它一般会与其他协议一同运行。比如RTSP(实时流传输协议)它就同时包含了RTP，RTCP和RTSP协议在里...	Biao 4842
使用libjrtp库,基于tcp传输H264码流,使用C+++新增_jrtpIib tcp_音视频 ... 以下是一个简单的C++程序,使用libjrtp库将H264码流通过TCP传输: "" include include <netinet/in.h> include <arpa/inet.h> include "jrtpIib3/rtppacket.h" i...	9-2
使用JrtpIib打包发送h264数据,关使用vlcIplayer播放 前段时间写了个测试程序,使用JrtpIib把h264打成标准的rtp数据包,并能通过vlc,mpalyer等播放器播放出来。这中间主要的难点是解析h264数据中的nal单元...	8-14
在Linux中编译JrtpIib 1.JrtpIib是什么 JrtpIib是一个基于C++、面向对象的RTP封装库，JrtpIib支持定义于RFC3550中的RTP协议，它使得发送和接收RTP报文变得异常简...	小蝴蝶的博客 1884
JrtpIib(RTP)传输H.264视频文件(2) 上一篇我们介绍了RTP协议的一些基本知识，下面我们介绍如何使用JrtpIib这个库传输H264编码。[cpp] view plaincopy J RTP传输: 好了，下面是我写的...	amduser的专栏 2124
qt5.70下编译APK使用JrtpIib传输H264文件，使用VLC直播视频 该工程是本人第一个qt工程，使用qt5.7编译的apk文件，直接通过qI来编译JrtpIib，压缩文档中包含了完整的qt工程，可以直接编译，调试和测试直播，...	06-29
使用JrtpIib发送流媒体数据Sender 使用JrtpIib发送流媒体数据Sender 主要通过JrtpIib进行音视频对应的H264文件的传输。传输到接收端后，接收端可使用ffmepg进行编解码播放	01-03
pingtaijianshe.rar_H.264_H.264 解码_JrtpIib h.264编解码平台，可直接运行的.文件大小20M	09-14
将H264码流打成RTP包后UDP发送-vc播放 将H264码流打成RTP包后，不借助第3方库，直接用UDP发送出去，vlc测试播放	07-18
JrtpIib 打包做了哪些事_使用JrtpIib打包发送h264数据，关使用vlcIplayer播放 步学园... 前段时间写了个测试程序，使用JrtpIib，把h264打成标准的rtp数据包，并能通过vlc，mpalyer等播放器播放出来。这中间主要的难点是解析h264数据中的n...	weixin_39762478的博客 129
linux 使用JrtpIib收发h.264视频文件 之前介绍过自己直接使用RTP收发h.264数据，这样有一个麻烦就是RTP协议的各个参数需要自己一个一个位的去填充，这样不利于发送也不方便接收，jr...	Biao 6521
使用JrtpIib打包发送h264数据，关使用vlcIplayer播放 前段时间写了个测试程序，使用JrtpIib，把h264打成标准的rtp数据包，并能通过vlc，mpalyer等播放器播放出来。这中间主要的难点是解析h264数据中的n...	xyyangkun的专栏 6564
使用JrtpIib传输H.264视频文件(3) 介绍如何发送h264，下面我们介绍下如何接收h264文件。其中主要关注的就是被拆分的数据包的重组，下面的代码中有详尽的注释。[cpp] view plaincop...	amduser的专栏 1386
JrtpIib收发程序 最新发布 ## 回答1: JrtpIib是一个开源的RTP协议实现库，用于音视频数据传输。这个库提供了接口简单易用，开发人员可以利用它来开发音视频传输应用程序。...	06-23

辟谣

土豆西瓜芝麻

码龄15年

暂无认证

232

2万+

5783

171万+

原创

周排名

总排名

访问

等级

1万+

861

1531

382

7230

积分

粉丝

获赞

评论

收藏

私信

关注

linux使用jrtpIib收发h264数据 解决了你的问题么？
可以写篇文章记录加深印象~

写文章

搜博文文章

热门文章

在本地shell脚本中ssh到远程服务器并执行命令 106927

Ubuntu 16.04安装docker详细步骤 97161

vs code中项目的基本配置—including路径、运行参数、debug配置 34978

微信小程序开发入门 (一) 31373

ubuntu 16.04下安装和配置ros 30081

最新评论

linux文件读写过程-各种层级结构

whtss: page cache 就是page cache，写成磁盘告诉缓存干什么？

逻辑地址、线性地址、物理地址的关系以... qq_36534753: 写的很棒，全看完了

C++中控制函数调用超时

Ezreal乐格: 在吗？但是最开始的线程并没有停止啊！

Posix共享内存

along_j: 想咨询一个问题：为什么父子进程共享内存那边的打印不是父一条子一条...

ROS的CMakeList文件详解

大白是无敌的: 也不知道哪里写的好的，收藏的人多，我也跟风风

您愿意向朋友推荐“博客详情页”吗？

强烈不推荐

不推荐

一般般

推荐

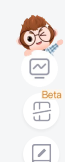
强烈推荐

最新文章

FFmpeg 解码内存泄漏汇总

sws_getContext函数无法释放问题

git submodule 使用教程



2023年 16篇	2022年 85篇
2021年 196篇	2020年 189篇
2019年 92篇	2018年 54篇
2011年 1篇	

“相关推荐”对你有帮助？

- 非常没帮助
- 没帮助
- 一般
- 有帮助
- 非常有帮助

[关于我们](#) [招](#)

阅读终点，创作起航，您可以撰写心得或摘录文章要点写篇博文。[去创作](#) × 22:00



土豆西瓜大芝麻

[关注](#)



0



2



0



[专栏目录](#)



[举报](#)

