

JT同学

铜牌4年

暂无认证

105

2万+

91万+

46万+

原创

周排名

总排名

访问

等级

5440

1861

1016

346

3945

积分

粉丝

获赞

评论

收藏



私信

关注

搜索

搜索

热门文章

从零开始写一个RTSP服务器（一）RTSP协议讲解

57102

C++ deque的用法与示例

38004

Linux I2C驱动框架（超详细）

35491

我的开源项目-RtspServer

25554

从零开始写一个RTSP服务器（二）RTSP协议的实现

21860

最新评论

我的开源项目-RtspServer

kaifa3n: 我测试 h264_rtp_server test.h264 也会这样的问题？请问解决了没？ m...

使用mp4v2封装H.264或mp4最简单示例

热心兔斯基: 应该是播放器的原因。换个播放器试试

使用mp4v2封装H.264或mp4最简单示例

ai804233436: h264的标准中，好像默认的设置是90KHz。所以我理解timeScale...

使用mp4v2封装H.264或mp4最简单示例

ai804233436: 有遇到，不知道原因呢。

深入学习Linux摄像头（二）v4l2驱动框架

qq_43745897: 请问往缓存区里面采集图像数据的帧率是多少呢？请问如果缓存区理...

您愿意向朋友推荐“博客详情页”吗？

强烈推荐

推荐

一般般

不推荐

强烈不推荐

最新文章

深入浅出MySQL事务（二）MVCC的实现原理

2020年 4篇

2019年 95篇

深入浅出MySQL事务（一）事务隔离

2018年 7篇

深入浅出MySQL索引（二）InnoDB存储引擎的索引

从零开始写一个RTSP服务器（四）一个传输H.264的RTSP服务器

JT同学

于 2019-08-10 20:58:08 发布

14088 收藏 71

分类专栏:

从零开始写一个RTSP服务器

文章标签:

rtsp流媒体服务器音视频

从零开始写一个RTS...

专栏收录该内容

187 订阅

10 篇文章

订阅专栏

从零开始写一个RTSP服务器系列

★我的开源项目-RtspServer

从零开始写一个RTSP服务器（一）RTSP协议讲解

从零开始写一个RTSP服务器（二）RTSP协议的实现

从零开始写一个RTSP服务器（三）RTP传输H.264

从零开始写一个RTSP服务器（四）一个传输H.264的RTSP服务器

从零开始写一个RTSP服务器（五）RTP传输AAC

从零开始写一个RTSP服务器（六）一个传输AAC的RTSP服务器

从零开始写一个RTSP服务器（七）多播传输RTP包

从零开始写一个RTSP服务器（八）一个多播的RTSP服务器

从零开始写一个RTSP服务器（九）一个RTP OVER RTSP/TCP的RTSP服务器

从零开始写一个RTSP服务器（四）一个传输H.264的RTSP服务器

文章目录

从零开始写一个RTSP服务器（四）一个传输H.264的RTSP服务器

- 一、建立套接字
- 二、接收客户端连接
- 三、解析命令
- 四、处理请求
 - 4.1 OPTIONS
 - 4.2 DESCRIBE
 - 4.3 SETUP
 - 4.4 PLAY
- 五、H.264 RTP打包发送
- 六、源码
 - h264_rtp_server.c
 - rtp.c
 - rtp.h
- 七、测试

本篇文章的目的：这篇文章的目的是实现一个播放H.264的RTSP服务器，相信如果是仔细看了前面几篇文章的话，这简直是易如反掌，就是把前面的东西拼到一起

所以这篇文章并不会讲述新的知识，只是把前面的东西拼凑到一起，整理一下思路，最后给出一个示例，下面开始讲解一下我提供的这个示例的运行流程

一、建立套接字

一开始进入main函数后，就监听服务器tcp套接字，绑定端口号，然后开始监听

然后再分别建立用于RTP和RTCP的udp套接字，绑定好端口

然后进入循环中开始服务

```
1 /*
2  * 作者: _JT_
3  * 博客: https://blog.csdn.net/weixin_42462282
4  */
5
6 main()
7 {
8     /* 创建服务器tcp套接字，绑定端口，监听 */
9     serverSockfd = createTcpSocket();
10    bindSocketAddr(serverSockfd, "0.0.0.0", SERVER_PORT);
11    listen(serverSockfd, 10);
12
13    /* 建立用于RTP和RTCP的udp套接字，绑定好端口 */
14    serverRtpSockfd = createUdpSocket();
15    serverRtcpSockfd = createUdpSocket();
```

二、接收客户端连接

在while循环中接收客户端，然后调用doClient服务

```
1 main()
2 {
3     ...
4     while(1)
5     {
6         clientSockfd = acceptClient(serverSockfd, clientIp, &clientPort);
7         doClient(clientSockfd, clientIp, clientPort, serverRtpSockfd, serverRtcpSockfd);
8     }
9 }
```

上面其实就是一个TCP服务器的基本步骤，没有什么特别的

下面来看一下doClient函数

三、解析命令

doClient就是一个while循环（这是一个同时只能服务一个客户的服务器），不断地接收命令解析命令，然后调用相应地操作

```
1 /*
2  * 作者: _JT_
3  * 博客: https://blog.csdn.net/weixin_42462282
4  */
5
6 doClient()
7 {
8     while(1)
9     {
10        recv(clientSockfd, rBuf, BUF_MAX_SIZE, 0);
11        ...
12        sscanf(line, "%s %s %s\r\n", method, url, version);
13        ...
14        sscanf(line, "CSeq: %d\r\n", &cseq)
15        ...
```

四、处理请求

在解析完客户端命令后，会调用相应的请求，处理完之后讲接收打印到sBuf中，然后发送给客户端

```
1 doClient()
2 {
3     while(1)
4     {
5         ...
6         /* 处理请求 */
7         if(!strcmp(method, "OPTIONS"))
8             handleCmd_OPTIONS(sBuf, cseq);
9         else if(!strcmp(method, "DESCRIBE"))
10            handleCmd_DESCRIBE(sBuf, cseq, url);
11        else if(!strcmp(method, "SETUP"))
12            handleCmd_SETUP(sBuf, cseq, clientRtpPort);
13        else if(!strcmp(method, "PLAY"))
14            handleCmd_PLAY(sBuf, cseq);
15    }
```

下面来看看各个请求的行动

4.1 OPTIONS

返回可用方法

```
1 /*
2  * 作者: _JT_
3  * 博客: https://blog.csdn.net/weixin_42462282
4  */
5
6 static int handleCmd_OPTIONS(char* result, int cseq)
7 {
8     sprintf(result, "RTSP/1.0 200 OK\r\n"
9                "CSeq: %d\r\n"
10                "Public: OPTIONS, DESCRIBE, SETUP, PLAY\r\n"
```

目录

从零开始写一个RTSP服务器（四）一个...

文章目录

一、建立套接字

二、接收客户端连接

三、解析命令

四、处理请求

4.1 OPTIONS

4.2 DESCRIBE

4.3 SETUP

4.4 PLAY

五、H.264 RTP打包发送

分类专栏

C/C++	1篇
STL源码剖析	18篇
MySQL	4篇
Linux内核	14篇
分布式	
nginx	
从零开始写一个RTSP服...	10篇
live555源码分析与应用	9篇
Linux驱动	19篇

```
11         "\r\n",
12         cseq);
13
14     return 0;
15 }
```

4.2 DESCRIBE

返回sdp文件信息，这是一个H.264的媒体描述信息，详细内容已经在前面文章讲解过，这里不再赘繁

```
1 /*
2  * 作者: _3T_
3  * 博客: https://blog.csdn.net/weixin_42462282
4  */
5
6 static int handleCmd_DESCRIBE(char* result, int cseq, char* url)
7 {
8     char sdp[500];
9     char localIp[100];
10
11     sscanf(url, "rtsp://%[*:]:", localIp);
12
13     sprintf(sdp, "v=0\r\n"
14             "o=- 99111 1 IN IP4 %s\r\n"
15             "s=-\r\n"
```

4.3 SETUP

SETUP过程发送服务端RTP端口和RTCP端口

```
1 /*
2  * 作者: _3T_
3  * 博客: https://blog.csdn.net/weixin_42462282
4  */
5
6 static int handleCmd_SETUP(char* result, int cseq, int clientRtpPort,
7                             int* localRtpSockfd, int* localRtcpSockfd)
8 {
9     sprintf(result, "RTSP/1.0 200 OK\r\n"
10             "CSeq: %d\r\n"
11             "Transport: RTP/AVP;unicast;client_port=%d-%d;server_port=%d-%d\r\n"
12             "Session: 66334873\r\n"
13             "\r\n",
14             cseq,
15             clientRtpPort,
```

4.4 PLAY

PLAY操作回复后，会开始发送RTP包

```
1 /*
2  * 作者: _3T_
3  * 博客: https://blog.csdn.net/weixin_42462282
4  */
5
6 static int handleCmd_PLAY(char* result, int cseq)
7 {
8     sprintf(result, "RTSP/1.0 200 OK\r\n"
9             "CSeq: %d\r\n"
10             "Range: npt=0.000-\r\n"
11             "Session: 66334873; timeout=60\r\n\r\n",
12             cseq);
13
14     return 0;
15 }
```

五、H.264 RTP打包发送

从H.264文件中读取一个NALU，向客户端发送RTP包（目的IP，目的RTP端口）

```
1 /*
2  * 作者: _3T_
3  * 博客: https://blog.csdn.net/weixin_42462282
4  */
5
6 doClient()
7 {
8     while(1)
9     {
10         ...
11         ...
12         ...
13         if(!strcmp(method, "PLAY"))
14         {
15             while(1)
```

下面看一看RTP打包过程。RTP打包实现了单NALU打包和分片打包

```
1 /*
2  * 作者: _3T_
3  * 博客: https://blog.csdn.net/weixin_42462282
4  */
5
6 static int rtpSendH264Frame(int socket, const char* ip, int16_t port,
7                             struct RtpPacket* rtpPacket, uint8_t* frame, uint32_t frameSize)
8 {
9     /* 如果包比较小，则采用单NALU打包 */
10     if (frameSize <= RTP_MAX_PKT_SIZE)
11     {
12         rtpSendPacket(socket, ip, port, rtpPacket, frameSize);
13     }
14     else //否则采用分片打包
15     {
```

六、源码

我也懒得建个Git仓了，源码就直接贴到这里吧，虽然有点长，嘻嘻

总共有3个文件，[h264_rtsp_server.c](#)、[rtp.c](#)、[rtp.h](#)

h264_rtsp_server.c

```
1 /*
2  * 作者: _3T_
3  * 博客: https://blog.csdn.net/weixin_42462282
4  */
5
6 #include <stdio.h>
7 #include <stdlib.h>
8 #include <stdint.h>
9 #include <string.h>
10 #include <sys/types.h>
11 #include <sys/socket.h>
12 #include <sys/socket.h>
13 #include <netinet/in.h>
14 #include <arpa/inet.h>
15 #include <time.h>
```

rtp.c

```
1 /*
2  * 作者: _3T_
3  * 博客: https://blog.csdn.net/weixin_42462282
4  */
5
6 #include <sys/types.h>
7 #include <sys/socket.h>
8 #include <arpa/inet.h>
9 #include <netinet/in.h>
10 #include <arpa/inet.h>
11
12 #include "rtp.h"
13
14 void rtpHeaderInit(struct RtpPacket* rtpPacket, uint8_t csrclen, uint8_t extension,
15                   uint8_t padding, uint8_t ver=1, uint8_t payloadType, uint8_t marker
```

rtp.h

```
1 /*
2  * 作者: _3T_
3  * 博客: https://blog.csdn.net/weixin_42462282
4  */
5
6 #ifndef _RTP_H_
7 #define _RTP_H_
8 #include <stdint.h>
9
10 #define RTP_VERSION 2
11
12 #define RTP_PAYLOAD_TYPE_H264 96
```

```
13 #define RTP_PAYLOAD_TYPE_AAC 97
14
15 #define RTP_HEADER_SIZE 12
```

七、测试

将三个文件保存下来，h264_rtsp_server.c、rtsp.c、rtsp.h

编译

```
1 # gcc h264_rtsp_server.c rtsp.c
```

运行，程序默认会打开test.h264的视频文件，如果你没有视频源的话，可以从RtspServer的example目录下获取

```
1 # ./a.out
```

运行后会打印一个url

```
1 | rtsp://127.0.0.1:8554
```

在vlc中输入url，即可看到视频

运行效果



- RTSP 服务器C语言 09-10
RTSP是实时流媒体传输协议。服务器和客户端之间通过RTSP协议实现握手和认证过程，通过RTP协议传输视频数据包。本资源通过C语言实现了RTSP服...
从头开始用C语言写一个简单rtsp转发服务器，支持H264,H265,流支持UDP和TCP jack_wu1992的博文 2148
最近在第一个rtsp的流媒体转发服务器，看了下ffmpeg，Gstream和live555，对于我的目标都不方便移植，所以从头写了一个简单的。网上找了不少资料...
- Live555实时视频流应用总结 marcosun_sw的博文 6728
live555实时流的流媒体服务器实现方法
- 基于C#的RTSP客户端简单协议程序 05-30
实现了简单的RTSP的客户端命令功能。 可以连接darwin服务器，并进行交互。实现了Options,Describe,Setup,Play,teardown命令。
- 基于h3531的live555例子（发送h264） 11-20
实现将已经编码的h264数据发送到网络（根据testh264VideoStreamer.cpp修改），一个线程将H264数据从文件中读取出来放到缓存 live555服务器不...
- rtsp协议c语言,RTSP协议 weixin_30069113的博文 648
RTSP简介RTSP(Real Time Streaming Protocol)是由Real Network和Netscape共同提出的如何有效地在IP网络上传输流媒体数据的应用层协议。RTSP对...
- c语言 rtsp协议,RTSP协议——第6章(6.4) weixin_36312271的博文 121
6.4Internal Sublayer Service provided within the MAC BridgeThe Internal Sublayer Service provided by a MAC entity to the MAC Relay Entity within a B...
- rtsp协议c语言,rtsp协议编程, TCP协议编程, c协议编程 weixin_36710660的博文 161
这篇文章主要从几个方面分析EasyCarmw的RTSP内容RTSP协议概述Wireshark抓包实例分析 一次完整RTSP的交互流程EasyCarmw项目代码中 RTSP的...
- c语言,rtsp客户端拉流,ffmpeg RTSP流拉取 weixin_36524882的博文 437
extern "C" {#include "libavformat/avformat.h"#include "libswscale/swscale.h"#include "libswresample/swresample.h"#include "libavutil/pixdesc.h"#includ...
- C# 音视频播放器（本地、rtsp等协议均支持） 03-22
这个是困扰我多日的一个问题。终于在茫茫网海中找到了最简便的解决办法 C#_player_rtsp 工程：.net 4.0 x86 winform 一个可以播放本地文件的播放器一...
- RTSP Server开源库测试 6-77
前2年搞过一个C#的RTSP Server项目，原项目中使用EasyIPCamera，这个项目目前已经改名为EasyRTSP Server，提供了一套完整的视频推送方案，这...
- Rtsp推送h264，Rtsp推送h265(hvc) lahenryfg的专栏 4493
rtsp推送相关的资料和测试软件比较少，本文介绍rtsp推送相关命令和测试结果。 1. rtsp推送流程 主要分两部分，第一部分先发送指令，第二部分发送r...
- 《ZLMediaKit源码学习笔记》（2）MediaServer作为RTSP转发服务器 最新发布 麦时小 2424
学习源码前，我们需要先熟悉如何使用ZLMediaKit提供的测试程序，正好本人手上有手机摄像头，本节就测试下如何使用MediaServer来转发摄像头的RT...
- RTSP服务端之——rtsp-server(轻量级RTSP / RTP流媒体服务器) Tinywan 6182
github: https://github.com/revenishcha/rtsp-server 轻量级RTSP / RTP流媒体服务器
- LIVE555再学习 – testh264VideoStreamer 源码分析 不聊废话，无以至千里 3133
上一篇文章我们已经讲了一部分：testh264VideoStreamer 重复从 H.264 基本流视频文件（名为test.264）中读取，并使用 RTP 多播进行流式传输。该...
- C#高性能RTSP播放器(使用FFmpeg封装自定义控件，使用简单) amo1988的专栏 866
vlc延时问题 使用vlc，即使优化参数，也有大概几百毫秒的延时string[] options = { "network-caching=300", "rtsp-tcp", "no-audio" };if ("network-cachin...
- live555 testh264VideoStreamer 运行错误(1) zhaoguangjun33的专栏 693
我在h35318上运行live555,出现以下的错误: 00:06:12 Groupsock(3: 232.87.34.182, 18888, 255): failed to join group: setsockopt(IP_ADD_MEMBERSH...
- rtsp流媒体服务器,基于RTP协议流媒体服务器的研究 weixin_42358261的博文 210
摘要: 流媒体技术的应用日益广泛,对流媒体技术的研究具有很大的实际意义.本文通过对RTP/RTCP协议的研究,分析流媒体服务器的一般功能和结构,给出...
- live555直播h264视频流 xushan2398的博文 878
刚开始接触live555，在live555testProgs中有很多很好的例子来讲解live555各方面的应用，但是都是以文件形式传到服务器中，然后广播出来的，但是很...
- java实现传输H.264的RTSP服务 冲的博文 702
java实现RTSP服务：H264文件初始化，封装成Rtp包。

“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

©2022 CSDN 皮肤主题：数字20 设计师：CSDN官方博客 返回顶部

关于我们 招聘纳士 商务合作 寻求报道 400-660-0108 kefu@csdn.net 在线客服 工作时间 8:30-22:00

公安备案号11010502030143 京ICP备10004658号 京公网安备11010502030143 经营性网站备案信息 北京互联网违法和不良信息举报中心 国家反诈中心 家长监护 网络110报警服务 中国互联网络信息中心 Chrome浏览器 隐私政策 版权与免责声明 版权申诉 出版物许可证 营业执照 1999-2022北京创新乐知网络技术有限公司

JT同学 关注

35 71 30 专栏目录