

1 RTSP服务器搭建

1 获取代码

2、强烈推荐

3、编译器

3.1、编译器版本要求

3.2、安装编译器

4、cmake

5、依赖库

5.1、依赖库列表

5.2、安装依赖库

6、构建和编译项目

7、运行

8 测试

推流测试

腾讯课堂 零声学院

FFmpeg/WebRTC/RTMP 音视频流媒体高级开发 <https://ke.qq.com/course/468797?tuin=137bb271>

使用的开源服务地址：<https://github.com/xiongziliang/ZLMediaKit>

1 获取代码

```
1 #国内用户推荐从同步镜像网站gitee下载
2 git clone --depth 1 https://gitee.com/xia-chu/ZLMediaKit.git
3 cd ZLMediaKit
4 #千万不要忘记执行这句命令
5 git submodule update --init
```

2、强烈推荐

如果你是位新手，强烈建议使用ubuntu16或更新版本编译ZLMediaKit，macOS是次选推荐平台，最不推荐的是centos6.*或windows平台。

3、编译器

3.1、编译器版本要求

ZLMediaKit采用了C++11的语法和库，要求编译器支持完整的C++11标准，亦即：

- linux上要求gcc 版本 ≥ 4.8 (4.7应该也能支持)
- macOS上clang \geq ??? (我也不知道，估计大部分不会遇到这个问题)
- windows 上visual studio ≥ 2015 (vs2013某些版本也能编译通过，如果怕麻烦建议直接vs2017)

3.2、安装编译器

- 如果你是debian系操作系统(包括ubuntu系用户)，一般自带的gcc版本够新，你可以这样安装gcc编译器：

```
1 sudo apt-get install build-essential
```

4、cmake

ZLMediaKit采用cmake来构建项目，通过cmake才能生成Makefile(或Xcode/VIS工程)，所以必须先安装cmake才能完成后续步骤。

- 如果你是debian系操作系统(包括ubuntu系用户)，一般自带的cmake版本够新，你可以这样安装cmake

```
1 sudo apt-get install cmake
```

5、依赖库

5.1、依赖库列表

ZLMediaKit可选依赖一些第三方库，这些库都不是必选的；在构建ZLMediaKit时，cmake能查找系统路径中的这些库，并根据安装情况选择是否开启相关特性，你可以选择安装这些依赖并启用相关特性：

- openssl
 - flash player在播放rtmp时，采用的是复杂握手模式，如果不安装该库，flash player将播放不了zlmediakit 提供的rtmp url.
 - 同时ZLMediaKit的https/rtsps相关功能需要使用openssl才能开启。

- ffmpeg
zlmmediakit可以通过fork ffmpeg进程的方式实现多种协议的拉流，编译时不需要安装FFmpeg。
- sdl、avcodec、avutil
这3个库供ZLMediaKit的test_player测试程序使用，你通常不需要安装这3个库。

5.2、安装依赖库

- Debian系(包括ubuntu) 系统下安装依赖的方法：

```
1 #除了openssl,其他其实都可以不安装
2 sudo apt-get install libssl-dev
3 sudo apt-get install libsdl-dev
4 sudo apt-get install libavcodec-dev
5 sudo apt-get install libavutil-dev
6 sudo apt-get install ffmpeg
```

6、构建和编译项目

- 在linux或macOS系统下,你应该这样操作：

```
1 cd ZLMediaKit
2 mkdir build
3 cd build
4 #macOS下可能需要这样指定openssl路径: cmake .. -DOPENSSL_ROOT_DIR=/usr/local/Cellar/openssl/1.0.2j/
5 cmake ..
6 make -j4
```

7、运行

ZLMediaKit工程主要生成3种二进制目标文件，他们的生成的路径在release目录下，这些目标文件主要分为：

- MediaServer进程这是ZLMediaKit作为服务器的主进程，该进程可以在免去开发的情况下直接作为测试流媒体服务器使用，如果你需要更复杂的业务逻辑，可以通过[Web HOOK](#)和[RESTful API](#)实现,同时你可以通过[配置文件](#)控制其参数。
 - 在linux下启动：

```
1 cd ZLMediaKit/release/linux/Debug
2 #通过-h可以了解启动参数
3 ./MediaServer -h
4 #以守护进程模式启动
5 ./MediaServer -d &
```

8 测试

推流测试

ZLMediaKit支持rtsp/rtmp/rtp推流，一般通常使用obs/ffmpeg推流测试，其中FFmpeg推流命令支持以下：

- 1、使用rtsp方式推流

```
1 # h264推流
2 ffmpeg -re -i "/path/to/test.mp4" -vcodec h264 -acodec aac -f rtsp
   -rtsp_transport tcp rtsp://127.0.0.1/live/test
3 # h265推流
4 ffmpeg -re -i "/path/to/test.mp4" -vcodec h265 -acodec aac -f rtsp
   -rtsp_transport tcp rtsp://127.0.0.1/live/test
```

- 2、使用rtmp方式推流

```
1 #如果未安装FFmpeg，你也可以用obs推流
2 ffmpeg -re -i "/path/to/test.mp4" -vcodec h264 -acodec aac -f flv
   rtmp://127.0.0.1/live/test
3 # RTMP标准不支持H265，但是国内有自行扩展的，如果你想让FFmpeg支持RTMP-H265，请
   按照此文章编译：https://github.com/ksvc/FFmpeg/wiki/hevcpush
```

- 3、使用rtp方式推流

```
1 # h264推流
2 ffmpeg -re -i "/path/to/test.mp4" -vcodec h264 -acodec aac -f rtp_
```

```
mpegts rtp://127.0.0.1:10000
```

```
3 # h265推流
```

```
4 ffmpeg -re -i "/path/to/test.mp4" -vcodec h265 -acodec aac -f rtp_
mpegts rtp://127.0.0.1:10000
```

文件(F) 编辑(E) 视图(V) 跳转(G) 捕获(C) 分析(A) 统计(S) 电话(Y) 无线(W) 工具(I) 帮助(H)

rtsp or rtsp or rtsp

No.	Time	Source	Destination	Protocol	Length	Info
7084	187.086109	111.229.231.225	192.168.2.225	RTSP	250	Reply: RTSP/1.0 200 OK
7085	187.086904	192.168.2.225	111.229.231.225	RTSP	239	SETUP rtsp://111.229.231.225:554/live/test/streamid=0 RTSP/1.0
7086	187.114813	111.229.231.225	192.168.2.225	RTSP	324	Reply: RTSP/1.0 200 OK
7087	187.115952	192.168.2.225	111.229.231.225	RTSP	239	SETUP rtsp://111.229.231.225:554/live/test/streamid=1 RTSP/1.0
7088	187.143527	111.229.231.225	192.168.2.225	RTSP	324	Reply: RTSP/1.0 200 OK
7089	187.144952	192.168.2.225	111.229.231.225	RTSP	188	RECORD rtsp://111.229.231.225:554/live/test RTSP/1.0
7091	187.180959	111.229.231.225	192.168.2.225	RTSP	312	Reply: RTSP/1.0 200 OK
7094	187.238928	192.168.2.225	111.229.231.225	RTCP	86	Sender Report
7095	187.239184	192.168.2.225	111.229.231.225	RTP	1478	PT=DynamicRTP-Type-96, SSRC=0x746EAD8, Seq=230, Time=757807685, Mark, Sender Report
7096	187.239185	192.168.2.225	111.229.231.225	RTP	557	PT=DynamicRTP-Type-96, SSRC=0x746EAD8, Seq=232, Time=757814885
7098	187.239250	192.168.2.225	111.229.231.225	RTP	106	PT=DynamicRTP-Type-96, SSRC=0x746EAD8, Seq=233, Time=757814885
7099	187.239344	192.168.2.225	111.229.231.225	RTP	1478	PT=DynamicRTP-Type-96, SSRC=0x746EAD8, Seq=234, Time=757814885, Mark, PT=DynamicRTP-
7100	187.239346	192.168.2.225	111.229.231.225	RTP	639	PT=DynamicRTP-Type-96, SSRC=0x746EAD8, Seq=236, Time=757811285
7101	187.239554	192.168.2.225	111.229.231.225	RTP	1478	PT=DynamicRTP-Type-96, SSRC=0x746EAD8, Seq=237, Time=757811285, Mark, PT=DynamicRTP-