

# 用 vlc 搭建简单流媒体服务器（UDP 和 TCP 方式）

这段时间用到了流媒体数据传输的 TCP 和 UDP 方式，感觉 vlc 可以做这方面的 demo，这里总结下，也方便我以后查阅。

## 简介

VLC 主页：<http://www.videolan.org>

VLC 不仅仅是一个播放器，也可以作为流媒体服务器使用。这个网上有相关的资料，我就不多说了，声明下本文用的 VLC 版本：2.0.3

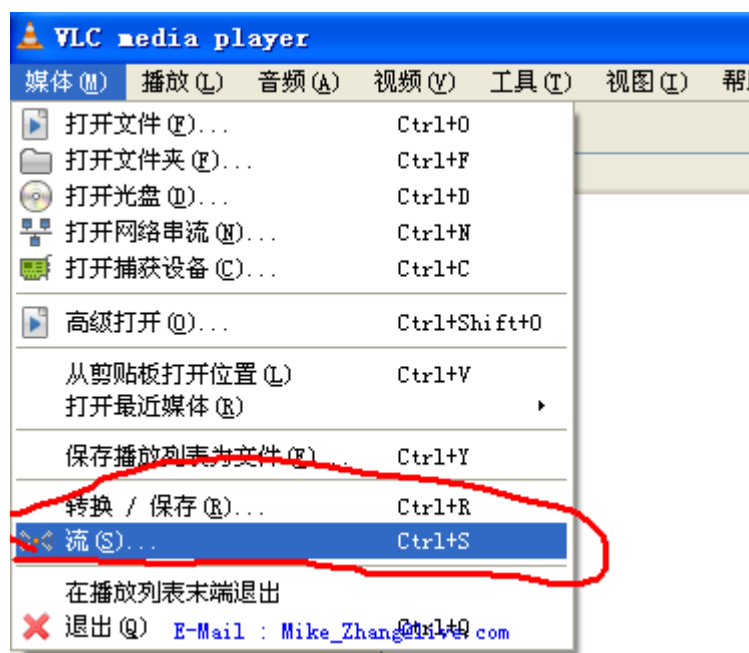
## 用 VLC 搭建基于 UDP 的流媒体服务器

流媒体服务器大多数是基于 UDP 的，这个在 VLC 中也有好几种实现，我这里只列出我用到的几个。

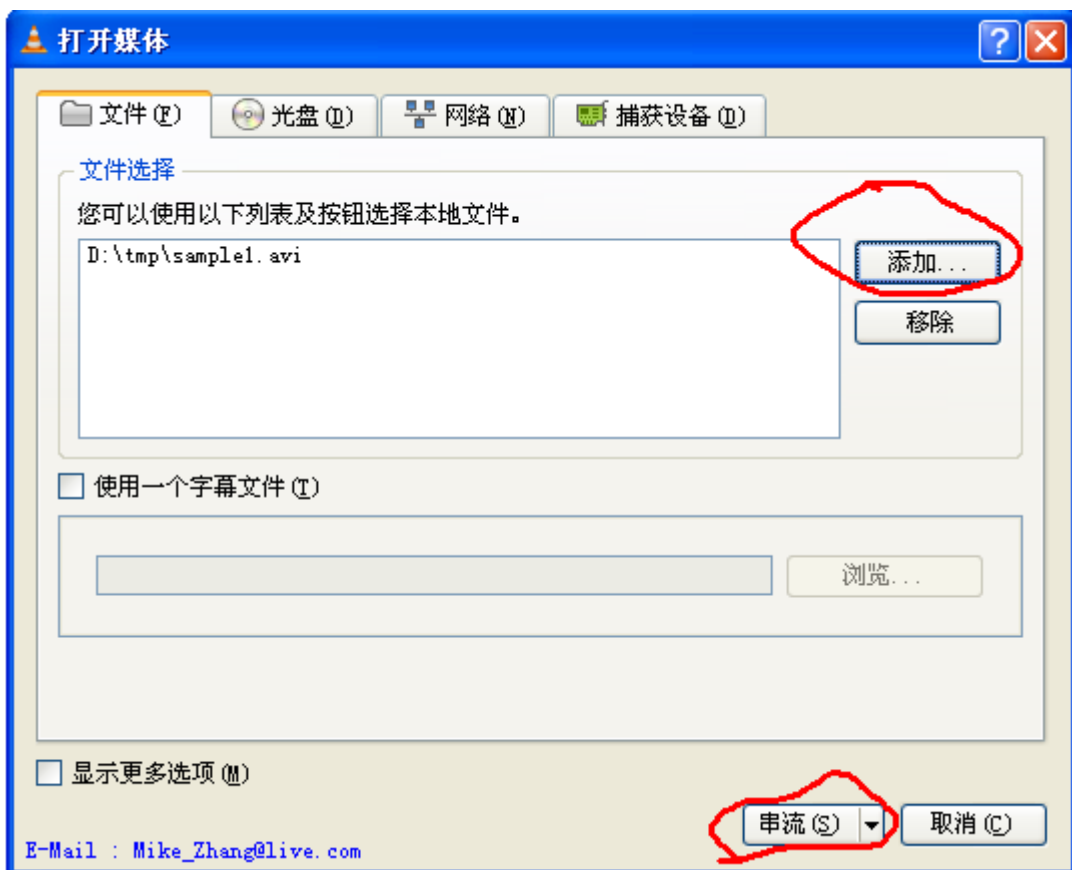
### 1、UDP（legacy）传统模式

#### 服务端操作

打开“媒体——流”选项，如图所示：



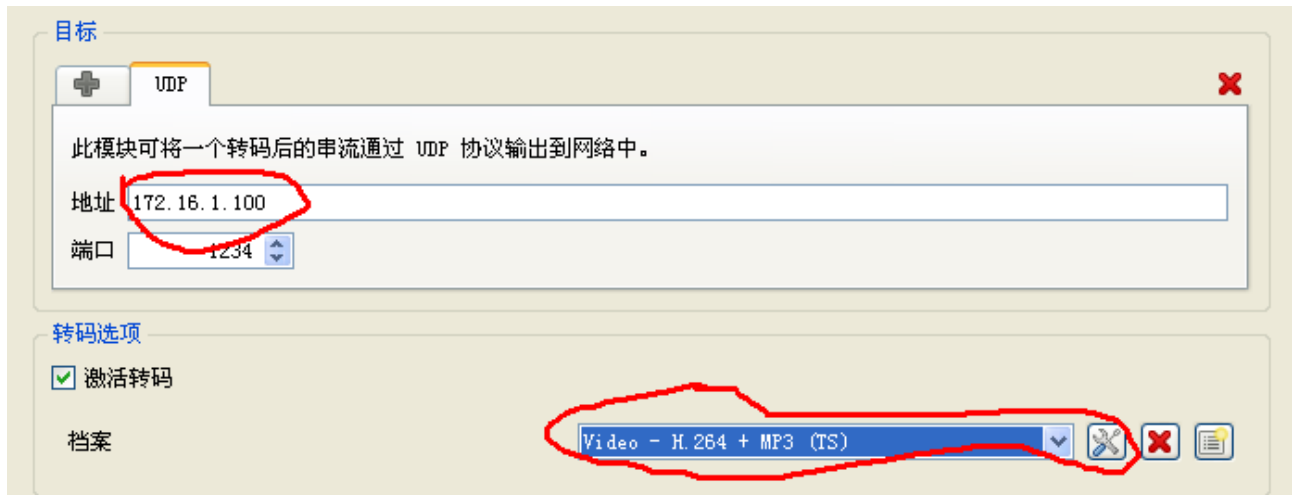
点击“添加”按钮选择流媒体文件，然后点击“串流”按钮



点击“下一个”后，在“目标设置”界面选择“UDP (legacy)”，然后点击添加按钮



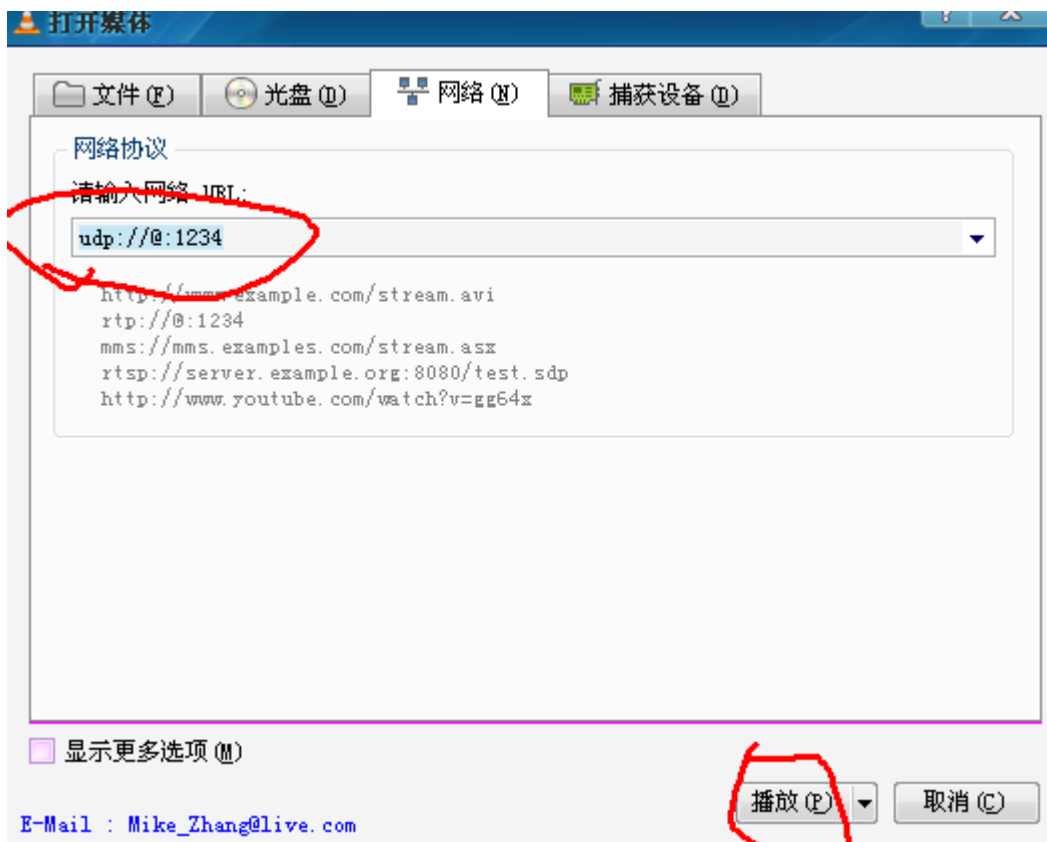
填上目标 ip 地址，并在“转码选项”中选择相应的编码类型（这个视情况而定）



可以点击“下一个”进行相关设置，也可以点击“串流”来完成服务端的设定。这里我点击“串流”完成服务端的设定。

## 客户端操作

点击“媒体——打开网络串流”，在弹出的框中输入“[udp://@1234](#)”，点击“播放”按钮即可。



通过界面操作比较简单，但如果经常进行此类工作的话，脚本是比较靠谱的。

## 命令行操作方式

服务端（172.16.1.1）：

```
vlc -vvv sample1.avi --sout udp:172.16.1.100:1234 --ttl 10
```

客户端（172.16.1.100）：

```
vlc udp://@:1234
```

## 2、RTP 方式

虽然有基于 TCP 的 RTP，但抓包发现，VLC 的 RTP 是基于 UDP 的。

| Filter: | Expression... | Clear      | Apply        |          |        |                  |
|---------|---------------|------------|--------------|----------|--------|------------------|
| No.     | Time          | Source     | Destination  | Protocol | Length | Info             |
| 677     | 18.482235     | 172.16.1.1 | 172.16.1.100 | UDP      | 1370   | Source port: top |
| 678     | 18.482256     | 172.16.1.1 | 172.16.1.100 | UDP      | 1370   | Source port: top |
| 679     | 18.482279     | 172.16.1.1 | 172.16.1.100 | UDP      | 1370   | Source port: top |
| 680     | 18.482303     | 172.16.1.1 | 172.16.1.100 | UDP      | 1370   | Source port: top |
| 681     | 18.482324     | 172.16.1.1 | 172.16.1.100 | UDP      | 1370   | Source port: top |
| 682     | 18.482348     | 172.16.1.1 | 172.16.1.100 | UDP      | 1370   | Source port: top |
| 683     | 18.482368     | 172.16.1.1 | 172.16.1.100 | UDP      | 1370   | Source port: top |
| 684     | 18.482389     | 172.16.1.1 | 172.16.1.100 | UDP      | 1370   | Source port: top |
| 685     | 18.482411     | 172.16.1.1 | 172.16.1.100 | UDP      | 1370   | Source port: top |
| 686     | 18.482433     | 172.16.1.1 | 172.16.1.100 | UDP      | 1370   | Source port: top |
| 687     | 18.482458     | 172.16.1.1 | 172.16.1.100 | UDP      | 1370   | Source port: top |
| 688     | 18.482482     | 172.16.1.1 | 172.16.1.100 | UDP      | 1370   | Source port: top |
| 689     | 18.482505     | 172.16.1.1 | 172.16.1.100 | UDP      | 1370   | Source port: top |
| 690     | 18.482528     | 172.16.1.1 | 172.16.1.100 | UDP      | 1370   | Source port: top |
| 691     | 18.482552     | 172.16.1.1 | 172.16.1.100 | UDP      | 1370   | Source port: top |
| 692     | 18.482574     | 172.16.1.1 | 172.16.1.100 | UDP      | 1370   | Source port: top |

<

Frame 685: 1370 bytes on wire (10960 bits), 1370 bytes captured (10960 bits)

Ethernet II, Src: Cadmusco\_00:24:73 (08:00:27:00:24:73), Dst: Cadmusco\_fa:5c:47 (08:00:27:00:5c:47)

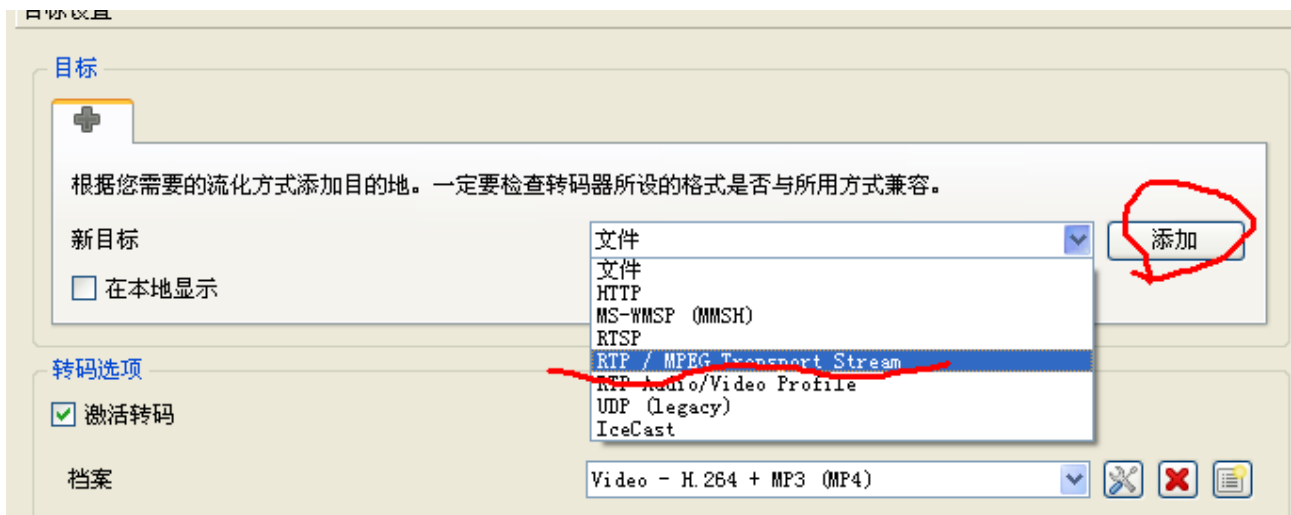
Internet Protocol Version 4, Src: 172.16.1.1 (172.16.1.1), Dst: 172.16.1.100 (172.16.1.100)

User Datagram Protocol, Src Port: topflow (2885), Dst Port: avt-profile-1 (5004)

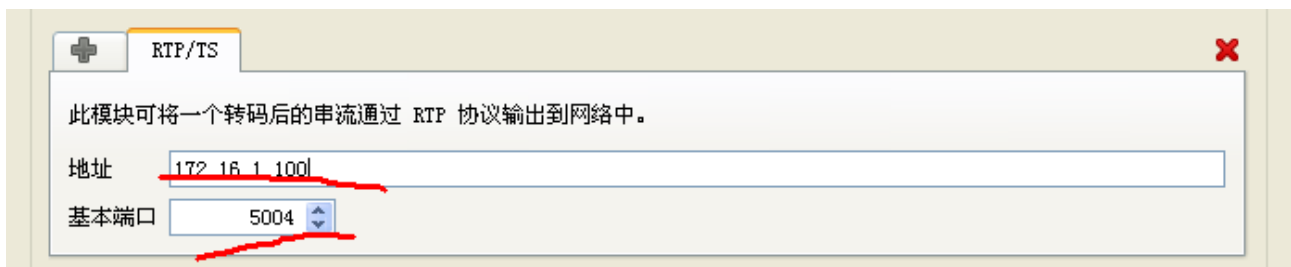
Data (1328 bytes)

## 服务端操作

前几步和“UDP（legacy）传统模式”类似，这里不再赘述。主要是在目标里面选择“RTP / MPEG Transport Stream”这个选项，如图所示：

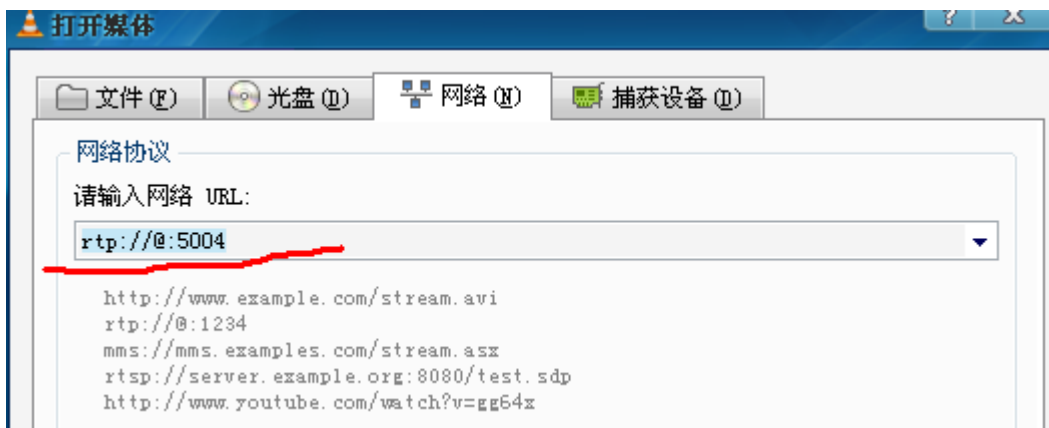


填 IP，选端口：



点击“串流”按钮即可完成服务端操作。

### 客户端操作



点击“播放”按钮即可。

### 命令行操作方式

服务端（172.16.1.1）：

```
vlc -vvv sample1.avi --sout
"#transcode{vcodec=h264,vb=0,scale=0,acodec=mpga,ab=128,channels=2,samplerate=44
```

E-Mail : [Mike\\_Zhang@live.com](mailto:Mike_Zhang@live.com)

```
100}:rtp{dst=172.16.1.100,port=5004,mux=ts,ttl=10}" -sout-keep
```

客户端 (172.16.1.100) :

```
vlc rtp://@:5004
```

### 3、RTSP 方式

RTSP 方式是通过 RTP 进行流媒体数据的传输的，VLC 的实现也是基于 UDP 的。这种方式网上的参考资料比较多，我就不截图了，直接叙述命令行方式的操作。

#### 命令行操作方式

服务端 (172.16.1.1) :

```
vlc -vvv sample1.avi --sout  
"#transcode{vcodec=h264,vb=0,scale=0,acodec=mpga,ab=128,channels=2,samplerate=44  
100}:rtp{sdp=rtsp://:8554/test}" --sout-keep
```

客户端 (172.16.1.100) :

```
vlc rtsp://172.16.1.1:8554/test
```

### 用 VLC 搭建基于 TCP 的流媒体服务器

VLC 的 http 方式搭建的流媒体服务器是用 TCP 方式进行数据传输的，这里就以这个为例子了。

#### HTTP 方式

这个的界面操作和 RTSP 的类似，就不赘述。

#### 命令行操作方式

服务端 (172.16.1.1) :

```
vlc -vvv sample1.avi --sout  
"#transcode{vcodec=h264,vb=0,scale=0,acodec=mpga,ab=128,channels=2,samplerate=44  
100}:http{mux=ffmpeg{mux=flv},dst=:8080/test}" --sout-keep
```

客户端 (172.16.1.100) :

```
vlc http://172.16.1.1:8080/test
```

## 附录

### 传统UDP 方式

服务端：

```
vlc -vvv sample1.avi --sout udp:172.16.1.100:54321 --ttl 10
```

客户端，监听本地端口：

```
vlc udp://@:54321
```

### RTP 方式

服务端：

```
vlc -vvv sample1.avi --sout  
"#transcode{vcodec=h264,vb=0,scale=0,acodec=mpga,ab=128,channels=2,samplerate=44  
100}:rtp{dst=172.16.1.100,port=5004,mux=ts,ttl=10}" --sout-keep
```

客户端：

```
vlc rtp://@:5004
```

### RTSP 方式

服务端：

```
vlc -vvv sample1.avi --sout  
"#transcode{vcodec=h264,vb=0,scale=0,acodec=mpga,ab=128,channels=2,samplerate=44  
100}:rtp{sdp=rtsp://:8554/test}" --sout-keep
```

客户端：

```
vlc rtsp://172.16.1.1:8554/test
```

### HTTP 方式

服务端：

```
vlc -vvv sample1.avi --sout  
"#transcode{vcodec=h264,vb=0,scale=0,acodec=mpga,ab=128,channels=2,samplerate=44  
100}:http{mux=ffmpeg{mux=flv},dst=:8080/test}" --sout-keep
```

客户端：

```
vlc http://172.16.1.1:8080/test
```