

基于nginx的hls直播系统

原创 2013年06月22日 20:04:31

48342

之前写了一篇基于nginx的hls点播系统，本质上是把一个媒体文件做成m3u8索引，对应的文件都是提前做好放在服务器上的。

nginx充当的是个Http 服务器的角色，之所以说是基于nginx的，是因为它可以支持实时。本文主要是描述一个直播系统，核心在于m3u8和里面对应的ts链接都是实时的可以刷新。类似于cntv里面的直播。

这里分按顺序分几个部分讲述：软件编译，rtmp源的提供，nginx配置，html代

1, 软件编译：

所需模块：nginx-rtmp-module

github:

<https://github.com/arut/nginx-rtmp-module#example-nginxconf>

这个模块对nginx的版本好像没有什么要求，我用1.2.2是可以的。编译方法github上写的很清楚。

```
[plain]
1. ./configure --add-module=<path-to-nginx-rtmp-module>
2. make
3. make install
```

1.3.14-1.5.0版本

```
[plain]
1. ./configure --add-module=<path-to-nginx-rtmp-module> --with-http_ssl_module
```

2, rtmp源的提供

一类是用一个已有的媒体文件，一类是用摄像头和麦克风采集。

例如：

```
[plain]
1. ffmpeg.exe -re -i sample.flv -vcodec copy -acodec copy -f flv rtmp://server-ip-address/hls/mystream
2. ffmpeg.exe -f dshow -i video="USB2.0 Camera" -vcodec libx264 -pix_fmt yuv420p -f flv rtmp://server-ip-address/hls/mystream
```

第一个是基于一个媒体文件的，必须用re，标识native frame rate，意思是按照播放的帧率。

第二个是基于dshow的，在windows上，编码用x264,图像用420p,

两种方式都是以rtmp协议发给server，其中hls和mystream各有含义。hls表示application，mystream表示一个实例。稍后解释。

3, nginx配置

这个nginx-rtmp-module里面已经包含了一个nginx.conf,位于test目录下，如果你已经有了一个nginx配置文件，那么只需要用



原创 39 粉丝 59 喜欢 1

等级： **博客 5** 访问量： 40. 积分： 2815 排名： 1.51万

广告

他的最新文章

- weppy模板功能
- python+Selenium2+chrome构建网页爬虫工具
- gststreamer 在visual studio 中环境录
- nginx结合moinmoin搭建wiki站点
- nginx+weppy配置灵活限速(pythor

文章分类

- DSP
- ffmpeg
- H.264
- 收藏的博客
- linux大类
- python-web

展开

文章存档

- 2013年7月
- 2013年6月
- 2013年5月
- 2013年3月
- 2012年12月
- 2012年9月

展开

他的热门文章

- svn回到历史的某个版本 110163
- 基于nginx的hls直播系统 48318
- python+Selenium2+chrome构建网页爬虫工具 44788
- nginx上搭建HLS流媒体服

即可包含这个新配置，Include必须与现有配置同级，即http级别的。

这里通常会有些问题，例如rtmp不能识别。

```
[plain]
1. unknown directive "?rtmp
2. unknown directive "rtmp" in /etc/nginx/conf.d/rtmp.conf:1
```

解决方法一般是两种，一个是新conf的编码必须是和原有的一样，一般都是ASCII，用file指令就知道。

一是重新编译后的nginx的model没有加载进去，可以尝试stop nginx再start就行。

```
[plain]
1. rtmp {
2.     server {
3.         listen 1935;
4.
5.         application myapp {
6.             live on;
7.
8.             #record keyframes;
9.             #record_path /tmp;
10.            #record_max_size 128K;
11.            #record_interval 30s;
12.            #record_suffix .this.is.flv;
13.
14.            #on_publish http://localhost:8080/publish;
15.            #on_play http://localhost:8080/play;
16.            #on_record_done http://localhost:8080/record_done;
17.
18.        }
19.        application hls {
20.            live on;
21.            hls on;
22.            hls_path /tmp/app;
23.            hls_fragment 5s;
24.
25.        }
26.    }
27. }
28. }
29.
30. http {
31.     server {
32.         listen 8080;
33.
34.         location /stat {
35.             rtmp_stat all;
36.             rtmp_stat_stylesheet stat.xsl;
37.         }
38.
39.         location /stat.xsl {
```

阅读全文

ffmpeg: 'UINT64_C' was not dec
n this scope

12163

jwplayer进阶HTML5

11685

nginx+webpy+uwsgi+jwplayer组
流媒体服务器

11685

如何方便的查看某个网站的服务器
web架构

10479

ffmpeg中获取字节流格式的两种方

9452

linux shell sleep/wait

8894

联系我们



请扫描二维码联系

webmaster@c

400-660-010

QQ客服 客服

关于 · 招聘 · 广告服务 · 百

©1999-2018 CSDN版权所有

京ICP证09002463号

经营性网站备案信息

网络110报警服务

中国互联网举报中心

北京互联网违法和不良信息举报中心



目前您尚未登录，请 [登录](#) 或 [注册](#) 后进行评论



duoheng001 2016-06-22 14:31

回复

11楼

不能生成m3u8 文件确定rtmp视频流，有一样问题的吗？



u010942420 2016-03-15 20:52

回复

10楼

作者你好，请问stat页面能不能显示m3u8的play状态？如果不能，你是怎么监测m3u8的客户端在线状况的？



i0924520 2015-01-16 14:59

1条回复

回复

9楼

为什么没有生成m3u8文件啊？