参考：http://blog.csdn.net/kl222/article/details/12886661

http://blog.chinaunix.net/uid-26000296-id-4335063.html

实验目的:

让Nginx支持flv和mp4格式文件，支持RTMP协议的直播和点播；

同时打开RTMP的HLS功能

资料:

HTTP Live Streaming（缩写是 HLS）是一个由苹果公司提出的基于HTTP的流媒体网络传输协议。

HLS只请求基本的HTTP报文，与实时传输协议（RTP)不同，HLS可以穿过任何允许HTTP数据通过的防火墙或者代理服务器。

它也很容易使用内容分发网络来传输媒体流。

使用ffmpeg来完成对flv、mp4、mp3等格式的转化（点播实验暂时不测试）

# **准备工作**

Nginx: http://nginx.org

模块:nginx\_mod\_h264\_streaming(支持h264编码的视频)

http://h264.code-shop.com/trac/wiki/Mod-H264-Streaming-Nginx-Version2

D:\项目\视频直播点播\nginx\_rtmp\nginx\_mod\_h264\_streaming-2.2.7.tar

模块:http\_flv\_module 支持flv nginx源码自带，不用额外下载

http://tengine.taobao.org/nginx\_docs/cn/docs/http/ngx\_http\_flv\_module.html

模块:http\_mp4\_module 支持mp4 nginx源码自带，不用额外下载

http://tengine.taobao.org/nginx\_docs/cn/docs/http/ngx\_http\_mp4\_module.html

下载地址:

http://h264.code-shop.com/download/nginx\_mod\_h264\_streaming-2.2.7.tar.gz

https://github.com/arut/nginx-rtmp-module

D:\项目\视频直播点播\nginx\_rtmp\ nginx-rtmp-module-master

1、安装依赖包:

#yum -y install gcc glibc glibc-devel make nasm pkgconfig lib-devel openssl-devel expat-devel gettext-devel libtool mhash.x86\_64 perl-Digest-SHA1.x86\_64

## 安装相关工具包

1). git

# mkdir soft-source

# cd soft-source

# wget http://codemonkey.org.uk/projects/git-snapshots/git/git-latest.tar.xz

# xz -d git-latest.tar.xz

# tar xzvf git-latest.tar

# cd git-2014-06-27

# autoconf

# ./configure

# make && make install

# git --version

git version 2.0.0.GIT

# cd ..

2). Zlib——机器默认安装了zlib-1.2.3,暂不安装

# wget http://zlib.net/zlib-1.2.8.tar.gz

# tar -zxvf zlib-1.2.8.tar.gz cd zlib-1.2.8

# ./configure

# make

# make install

# cd ..

3). pcre——机器默认安装了pcre-7.8,暂不安装

# wget http://exim.mirror.fr/pcre/pcre-8.12.tar.gz

# tar zxvf pcre-8.12.tar.gz

# cd pcre-8.12

# ./configure

# make && make install

# cd ..

4). yadmi

yadmi的作用是为flv文件添加关键帧，才能实现拖动播放

# wget http://sourceforge.net/projects/yamdi/files/yamdi/1.4/yamdi-1.4.tar.gz/download

# tar xzvf yamdi-1.4.tar.gz

# cd yamdi-1.4

# make && make install

# cd ..

使用方法：

# yamdi -i input.flv -o out.flv

给input.flv文件 添加关键帧，输出为out.flv文件

5). OpenSSL——机器默认安装了openssl-1.0,暂不安装

# wget http://www.openssl.org/source/openssl-1.0.1c.tar.gz

# tar -zxvf openssl-1.0.1c.tar.gz

# ./config

# make

# make install

3、安装ffmpeg及其依赖包：

1). Yasm

# wget http://www.tortall.net/projects/yasm/releases/yasm-1.2.0.tar.gz

# tar xzvf yasm-1.2.0.tar.gz

# cd yasm-1.2.0

# ./configure

# make

# make install

# cd ..

2). x264

# git clone git://git.videolan.org/x264

# cd x264

# ./configure --enable-shared

# make

# make install

# cd ..

3). LAME

# wget http://downloads.sourceforge.net/project/lame/lame/3.99/lame-3.99.5.tar.gz

# tar xzvf lame-3.99.5.tar.gz

# cd lame-3.99.5

#./configure --enable-nasm

# make

# make install

# cd ..

4). libogg

# wget http://downloads.xiph.org/releases/ogg/libogg-1.3.0.tar.gz

# tar xzvf libogg-1.3.0.tar.gz

# cd libogg-1.3.0

# ./configure

# make

# make install

# cd ..

5). libvorbis

# wget http://downloads.xiph.org/releases/vorbis/libvorbis-1.3.3.tar.gz

# tar xzvf libvorbis-1.3.3.tar.gz

# cd libvorbis-1.3.3

# ./configure

# make

# make install

# cd ..

6). libvpx

# git clone http://git.chromium.org/webm/libvpx.git

# cd libvpx

# ./configure --enable-shared

# make

# make install

# cd ..

7). FAAD2

# wget http://downloads.sourceforge.net/project/faac/faad2-src/faad2-2.7/faad2-2.7.tar.gz

# tar zxvf faad2-2.7.tar.gz

# cd faad2-2.7

# ./configure

# make

# make install

# cd ..

8). FAAC

# wget http://downloads.sourceforge.net/project/faac/faac-src/faac-1.28/faac-1.28.tar.gz

# tar zxvf faac-1.28.tar.gz

# cd faac-1.28

# ./configure

编译FAAC-1.28时遇到错误：

mpeg4ip.h:126: error: new declaration ‘char\* strcasestr(const char\*, const char\*)’

解决方法：

从123行开始修改此文件mpeg4ip.h，到129行结束。

修改前：

#ifdef \_\_cplusplus

extern "C" {

#endif

char \*strcasestr(const char \*haystack, const char \*needle);

#ifdef \_\_cplusplus

}

#endif

修改后：

#ifdef \_\_cplusplus

extern "C++" {

#endif

const char \*strcasestr(const char \*haystack, const char \*needle);

#ifdef \_\_cplusplus

}

#endif

# make

# make install

# cd ..

9). Xvid

# wget http://downloads.xvid.org/downloads/xvidcore-1.3.2.tar.gz

# tar zxvf xvidcore-1.3.2.tar.gz

# cd xvidcore/build/generic

# ./configure

# make

# make install

# cd ..

10). ffmpeg

# git clone git://source.ffmpeg.org/ffmpeg

# cd ffmpeg

# ./configure --prefix=/opt/ffmpeg/ --enable-version3 --enable-libvpx --enable-libfaac --enable-libmp3lame --enable-libvorbis --enable-libx264 --enable-libxvid --enable-shared --enable-gpl --enable-postproc --enable-nonfree --enable-avfilter --enable-pthreads

# make && make install

# cd ..

修改/etc/ld.so.conf如下:

include ld.so.conf.d/\*.conf

/lib

/lib64

/usr/lib

/usr/lib64

/usr/local/lib

/usr/local/lib64

/opt/ffmpeg/lib

# ldconfig

## 安装Nginx相关模块

1. 模块安装

# wget http://h264.code-shop.com/download/nginx\_mod\_h264\_streaming-2.2.7.tar.gz

# tar zxvf nginx\_mod\_h264\_streaming-2.2.7.tar.gz

# git clone git://github.com/arut/nginx-rtmp-module.git

2. 创建用户和组新睿社区提供

# groupadd www

# useradd -g www www

3. nginx安装

# wget http://nginx.org/download/nginx-1.6.0.tar.gz

# tar zxvf nginx-1.6.0.tar.gz

# cd nginx-1.6.0

# **配置Nginx，实现VOD\LIVE**

1. 设置configure，nginx的补充编译，增加FLV和MP4功能。

# cd cd nginx-1.6.0

# vim nginx\_configure.sh

#!/bin/sh

echo "configure start ..."

./configure --prefix=/opt/nginx \

--add-module=.. /nginx\_mod\_h264\_streaming-2.2.7 \

--with-http\_flv\_module \

--with-http\_ssl\_module \

--with-http\_mp4\_module \

--with-http\_gzip\_static\_module \

--with-http\_stub\_status\_module \

--with-pcre=/home/cloudedu/nginx/tools/pcre-8.12 \

--with-zlib=/home/cloudedu/nginx/tools/zlib-1.2.8 \

--user=www --group=www \

--add-module=../nginx-rtmp-module-master \

--with-cc-opt=-I/opt/ffmpeg/include \

--with-ld-opt=`-L/opt/ffmpeg/lib -Wl, -rpath=/opt/ffmpeg/lib`

echo "configure end!"

【保存并退出】

# chmod +x nginx\_configure.sh

# ./nginx\_configure.sh

# make && make install

之后make，会报错【ngx\_http\_streaming\_module.c:158: 错误：‘ngx\_http\_request\_t’ 没有名为 ‘zero\_in\_uri’ 的成员】之类的错误。这需要修改nginx\_mod\_h264\_streaming-2.2.7的源代码：修改ngx\_http\_streaming\_module.c，注释掉 if (r->zero\_in\_uri) { return NGX\_DECLINED; } 这一段。 之后make clean，之后重新configure，重新make，之后make install。

http\_flv\_module和http\_mp4\_module即为对应的解析和seek功能支持。

2. conf/nginx.conf 支持

# vim conf/nginx.conf

【编辑nginx.conf】用nginx.conf替换默认的.

【保存并退出】

# **脚本部署**

将目录livenotesh拷贝到/opt/目录下，给改目录下sh增加执行权限。

新睿社区提供