Hướng dẫn cài đặt Nginx RTMP Streaming

* Ngày này nhu cầu sử dụng các dịch vụ streamming trực tuyến như Twitch.tv hoặc Ustream là rất lớn, để truyền tải video tới người xem. Nhưng đôi khi bạn muốn kiểm soát kênh stream của mình hoặc muốn người khác có thể phát trực tuyến với bạn. Hướng dẫn sau sẽ trình bày các vấn đề cơ bản về việc thiết lập máy chủ RTMP cho giải pháp streamming.

**Contents**

* 1 1. Cài đặt Nginx và Nginx-RTMP
* 2 2. Cài đặt FFmpeg
* 3 3. Cấu hình Nginx-RTMP, FFmpeg và HLS
  + 3.1 3.1 Cấu hình Streaming Video
  + 3.2 3.2 Cấu hình Encoding Video
* 4 4. Dùng OpenBroadcaster để publish live video
  + - 4.0.1 Related

1. Cài đặt Nginx và Nginx-RTMP

– Cài đặt các gói thư viện cần thiết để biên dịch Nginx và Nginx-RTMP từ source

yum -y install gcc gcc-c++ make zlib-devel pcre-devel openssl-devel

– Thư mục chứa source code mặc định là /usr/local/src

cd /usr/local/src

– Tải về mã nguồn Nginx và Nginx-RTMP

wget http://nginx.org/download/nginx-1.12.1.tar.gz

wget https://github.com/arut/nginx-rtmp-module/archive/master.zip

– Giải nén mã nguồn vừa tải về

nginxVersion="1.12.1"

tar -xzf nginx-$nginxVersion.tar.gz

unzip master

ln -sf nginx-$nginxVersion nginx # Thuận lợi cho việc upgrade version nginx sau này.

– Chuyển vào thư mục chứa mã nguồn Nginx và tiến hành biên dịch

cd nginx

– Tạo Makefile để Nginx sẽ được biên dịch với module Nginx-RTMP

./configure \

--user=nginx \

--group=nginx \

--prefix=/etc/nginx \

--sbin-path=/usr/sbin/nginx \

--conf-path=/etc/nginx/nginx.conf \

--pid-path=/var/run/nginx.pid \

--lock-path=/var/run/nginx.lock \

--error-log-path=/var/log/nginx/error.log \

--http-log-path=/var/log/nginx/access.log \

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

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

--with-http\_ssl\_module \

--with-pcre \

--with-file-aio \

--with-http\_realip\_module \

--without-http\_scgi\_module \

--without-http\_uwsgi\_module \

--without-http\_fastcgi\_module \

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

--add-module=../nginx-vod-module-1.13

– Tiến hành biên dịch và cài đặt

make

make install

– Tạo user nginx to the system:

useradd -r nginx

– Nội dung file

/etc/init.d/nginx:

#!/bin/sh

#

# nginx - this script starts and stops the nginx daemin

#

# chkconfig: - 85 15

# description: Nginx is an HTTP(S) server, HTTP(S) reverse \

# proxy and IMAP/POP3 proxy server

# processname: nginx

# config: /etc/nginx/nginx.conf

# pidfile: /var/run/nginx.pid

# user: nginx

# Source function library.

. /etc/rc.d/init.d/functions

# Source networking configuration.

. /etc/sysconfig/network

# Check that networking is up.

[ "$NETWORKING" = "no" ] && exit 0

nginx="/usr/sbin/nginx"

prog=$(basename $nginx)

NGINX\_CONF\_FILE="/etc/nginx/nginx.conf"

lockfile=/var/run/nginx.lock

start() {

[ -x $nginx ] || exit 5

[ -f $NGINX\_CONF\_FILE ] || exit 6

echo -n $"Starting $prog: "

daemon $nginx -c $NGINX\_CONF\_FILE

retval=$?

echo

[ $retval -eq 0 ] && touch $lockfile

return $retval

}

stop() {

echo -n $"Stopping $prog: "

killproc $prog -QUIT

retval=$?

echo

[ $retval -eq 0 ] && rm -f $lockfile

return $retval

}

restart() {

configtest || return $?

stop

start

}

reload() {

configtest || return $?

echo -n $"Reloading $prog: "

killproc $nginx -HUP

RETVAL=$?

echo

}

force\_reload() {

restart

}

configtest() {

$nginx -t -c $NGINX\_CONF\_FILE

}

rh\_status() {

status $prog

}

rh\_status\_q() {

rh\_status >/dev/null 2>&1

}

case "$1" in

start)

rh\_status\_q && exit 0

$1

;;

stop)

rh\_status\_q || exit 0

$1

;;

restart|configtest)

$1

;;

reload)

rh\_status\_q || exit 7

$1

;;

force-reload)

force\_reload

;;

status)

rh\_status

;;

condrestart|try-restart)

rh\_status\_q || exit 0

;;

\*)

echo $"Usage: $0 {start|stop|status|restart|condrestart|try-restart|reload|force-reload|configtest}"

exit 2

esac

– Tạo Nginx init scripts

wget -O /etc/init.d/nginx https://gist.github.com/sairam/5892520/raw/b8195a71e944d46271c8a49f2717f70bcd04bf1a/etc-init.d-nginx

chmod +x /etc/init.d/nginx

chkconfig --add nginx

chkconfig --level 345 nginx on

– Khởi động Nginx

service nginx start

– T/H: Upgrade ( cài lại phiên bản khác) version Nginx:

nginxVersion="1.12.0" # set the value here from nginx website

wget http://nginx.org/download/nginx-$nginxVersion.tar.gz

tar -xzf nginx-$nginxVersion.tar.gz

rm nginx # removes the soft link

ln -sf nginx-$nginxVersion nginx

cd nginx

Rồi thực hiện cài đặt lại theo các bước ở trên

2. Cài đặt FFmpeg

– Tạo file repo /etc/yum.repos.d/dag.repo

vi /etc/yum.repos.d/dag.repo

– Thêm nội dung sau vào file sau đó lưu lại

[dag]

name=DAG RPM Repository

baseurl=http://apt.sw.be/redhat/el$releasever/en/$basearch/dag

gpgcheck=1

enabled=1

– Để sử dụng DAG repository với công cụ yum, cần phải thêm DAG’s GPG key. Sử dụng lệnh sau để thêm GPG key cho DAG.

rpm --import http://apt.sw.be/RPM-GPG-KEY.dag.txt

– Cài đặt FFmpeg

yum install ffmpeg ffmpeg-devel ffmpeg-libpostproc

Cách khác cài đặt, do cách trên lỗi:

**Cài đặt ffmpeg để chuyển đổi video thường sang dạng Streaming (ts)**

**Link: https://trac.ffmpeg.org/wiki/CompilationGuide/Centos**

( Kết hợp link trên và code phía dưới, đã test cài ok. Một số lib ko cài đc theo link dưới thì lấy theo link bên trên )

Cài đặt các trình biên dịch cần thiết. **Lưu ý:** Tất cả các lệnh dưới đây phải chạy trên quyền cao nhất của hệ thống (root).

# rm -rf /etc/yum.repos.d/dag.repo

yum update

yum install -y autoconf automake cmake freetype-devel gcc gcc-c++ git libtool make mercurial nasm pkgconfig zlib-devel

**i1. Biên dịch yasm từ source**

mkdir ~/ffmpeg\_sources

cd ~/ffmpeg\_sources

git clone --depth 1 git://github.com/yasm/yasm.git

cd yasm

autoreconf -fiv

./configure --prefix="/opt/ffmpeg\_build" --bindir="$HOME/bin"

make

make install

make distclean

**i2. Biên dịch gói mã hóa video dạng x264 và x265**

cd ~/ffmpeg\_sources

git clone --depth 1 git://git.videolan.org/x264

cd x264

PKG\_CONFIG\_PATH="/opt/ffmpeg\_build/lib/pkgconfig"

./configure --prefix="/opt/ffmpeg\_build" --bindir="$HOME/bin" --enable-static

make

make install

make distclean

cd ~/ffmpeg\_sources

hg clone https://bitbucket.org/multicoreware/x265

cmake -G "Unix Makefiles" -DCMAKE\_INSTALL\_PREFIX="/opt/ffmpeg\_build" -DENABLE\_SHARED:bool=off source

**i3. Biên dịch gói mã hóa âm thanh chuẩn ACC**

cd ~/ffmpeg\_sources

git clone --depth 1 git://git.code.sf.net/p/opencore-amr/fdk-aac

cd fdk-aac

autoreconf -fiv

./configure --prefix="/opt/ffmpeg\_build" --disable-shared

make

make install

make distclean

**i4. Biên dịch gói mã hóa âm thanh LAME**

cd ~/ffmpeg\_sources

curl -L -O 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 --prefix="/opt/ffmpeg\_build" --bindir="$HOME/bin" --disable-shared --enable-nasm

make

make install

make distclean

**i5. Biên dịch code OPUS**

cd ~/ffmpeg\_sources

git clone http://git.opus-codec.org/opus.git

cd opus

autoreconf -fiv

./configure --prefix="/opt/ffmpeg\_build" --disable-shared

make

make install

make distclean

**i6. Biên dịch thư viện âm thanh OGG**

cd ~/ffmpeg\_sources

curl -O http://downloads.xiph.org/releases/ogg/libogg-1.3.2.tar.gz

tar xzvf libogg-1.3.2.tar.gz

cd libogg-1.3.2

./configure --prefix="/opt/ffmpeg\_build" --disable-shared

make

make install

make distclean

**i7. Biên dịch thư viện mã hóa âm thanh Vorbis**

cd ~/ffmpeg\_sources

curl -O http://downloads.xiph.org/releases/vorbis/libvorbis-1.3.4.tar.gz

tar xzvf libvorbis-1.3.4.tar.gz

cd libvorbis-1.3.4

LDFLAGS="-L/opt/ffmeg\_build/lib" CPPFLAGS="-I/opt/ffmpeg\_build/include" ./configure --prefix="/opt/ffmpeg\_build" --with-ogg="/opt/ffmpeg\_build" --disable-shared

make

make install

make distclean

**i8. Biên dịch thư viện libvpx của WebM**

cd ~/ffmpeg\_sources

git clone --depth 1 https://chromium.googlesource.com/webm/libvpx.git

cd libvpx

./configure --prefix="/opt/ffmpeg\_build" --disable-examples

make

make install

make clean

**i9. Biên dịch ffmpeg**

cd ~/ffmpeg\_sources

git clone --depth 1 https://git.videolan.org/git/ffmpeg.git

cd ffmpeg

PKG\_CONFIG\_PATH="/opt/ffmpeg\_build/lib/pkgconfig" ./configure --prefix="/opt/ffmpeg\_build" --extra-cflags="-I/opt/ffmpeg\_build/include" --extra-ldflags="-L/opt/ffmpeg\_build/lib" --bindir="$HOME/bin" --pkg-config-flags="--static" --enable-gpl --enable-nonfree --enable-libfdk-aac --enable-libfreetype --enable-libmp3lame --enable-libopus --enable-libvorbis --enable-libvpx --enable-libx264

make

make install

make distclean

3. Cấu hình Nginx-RTMP, FFmpeg và HLS

– Tạo các thư mục để chứa dữ liệu streaming và recording.

mkdir /home/HLS

mkdir /home/HLS/live              # Chứa dữ liệu HLS streaming cho desktop

mkdir /home/HLS/mobile            # Chứa dữ liệu HLS streaming cho mobile

mkdir /home/video\_recordings      # Chứa video ghi hình

– Mở port trên firewall cho Nginx và RTMP

+ Mở file cấu hình iptables

vi /etc/sysconfig/iptables

+ Thêm nội dung sau vào file:

-A INPUT M state --state NEW m TCP P dport 80 J ACCEPT

-A INPUT M state --state NEW m TCP P dport 1935 J ACCEPT

+ Lưu lại. Dùng lệnh sau để nạp lại rule

service iptables save

3.1 Cấu hình Streaming Video

– Mở file cấu hình Nginx

vi /usr/local/nginx/conf/nginx.conf

– Thêm vào nội dung sau:

rtmp {

server {

listen 1935;                        # Port để Nginx chấp nhận kết nối RTMP

chunk\_size 4096;                   # Kích thước chunk tối đa để ghép kênh stream

allow play all;

# Tạo một luồng HLS streaming là “live” dành cho desktop

application live {

allow play all;                       # Cho phép khách xem video

live on;                                # Bật luồng streaming “live”

record all;                             # Bật ghi hình video và audio streaming

record\_path /home/video\_recordings;     # Thư mục chứa các file ghi hình flv

record\_unique on;   # Khi bật chế độ này sẽ thêm thời gian vào tên các file ghi hình

hls on;                                 # Bật HLS

hls\_nested on;                      # Bật chế độ nested. Ở chế độ này một thư mục con sẽ được tạo ra cho mỗi luồng stream. Các HLS playlist và fragment sẽ được tạo ra trong thư mục con đó

hls\_path /home/HLS/live;   # Thiết lập thư mục chứa HLS playlist và fragment

hls\_fragment 10s                # Thiết lập chiều dài HLS fragment

}

# Tạo một luồng HLS streaming là "mobile" dành cho mobile

application mobile {

allow play all;                       # Cho phép chơi video

live on;                               # Bật chế độ live

hls on;                                # Bật chế độ HLS

hls\_nested on;

hls\_path /home/HLS/mobile;

hls\_fragment 10s;

}

# Cho phép xem lại các video live streams đã được ghi hình bằng URL: rtmp:// 61.14.232.64:1935/vod/filename.flv

application vod {

play /home/video\_recordings;

}

}

}

server {

listen 80;

server\_name 61.14.232.64;

# Tạo một đường dẫn http cho HLS Streaming Video theo URL: "http:// 61.14.232.64/live/123host/index.m3u8"

location /live {

types {

application/vnd.apple.mpegurl m3u8;

}

alias /home/HLS/live;

add\_header Cache-Control no-cache;

}

location /mobile {

types {

application/vnd.apple.mpegurl m3u8;

}

alias /home/HLS/mobile;

add\_header Cache-Control no-cache;

}

location /stats {

stub\_status;

}

location / {

root   html;

index  index.html index.htm;

}

}

}

3.2 Cấu hình Encoding Video

– Mở file cấu hình nginx

vi /usr/local/nginx/conf/nginx.conf

– Sử dụng module FFmpeg để encoding các live stream video về định dạng .flv và lưu ở thư mục /home/video\_recordings. Ta có thể tùy chỉnh các thông số như bitrate video, bitrate audio và độ phân giải.

Định dạng như sau: rtmp://ip-server:port/application-name/stream-name

* -i: địa chỉ ứng dụng streaming.
* -b:v bitrate video
* -c:v bộ mã hóa hình ảnh
* -s: độ phân giải
* -f: định dạng xuất
* -bufsize: kích thước bộ đệm

– Thêm vào nội dung sau:

exec ffmpeg -i rtmp://61.14.232.64:1935/$app/$name -acodec copy -c:v libx264 -preset veryfast -profile:v baseline -vsync cfr -s 480x360 -b:v 400k maxrate 400k -bufsize 400k -threads 0 -r 30 -f flv rtmp://61.14.232.64:1935/mobile/$;

– Cuối cùng khởi động lại Nginx

service nginx restart

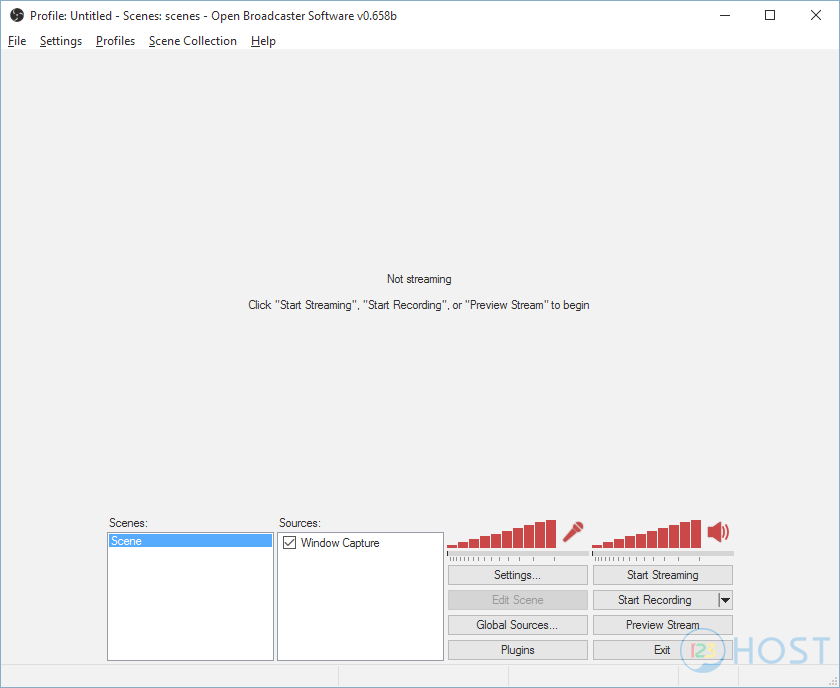
*–* Kiểm tra xem nginx và module rtmp chạy chưa

netstat -nltp

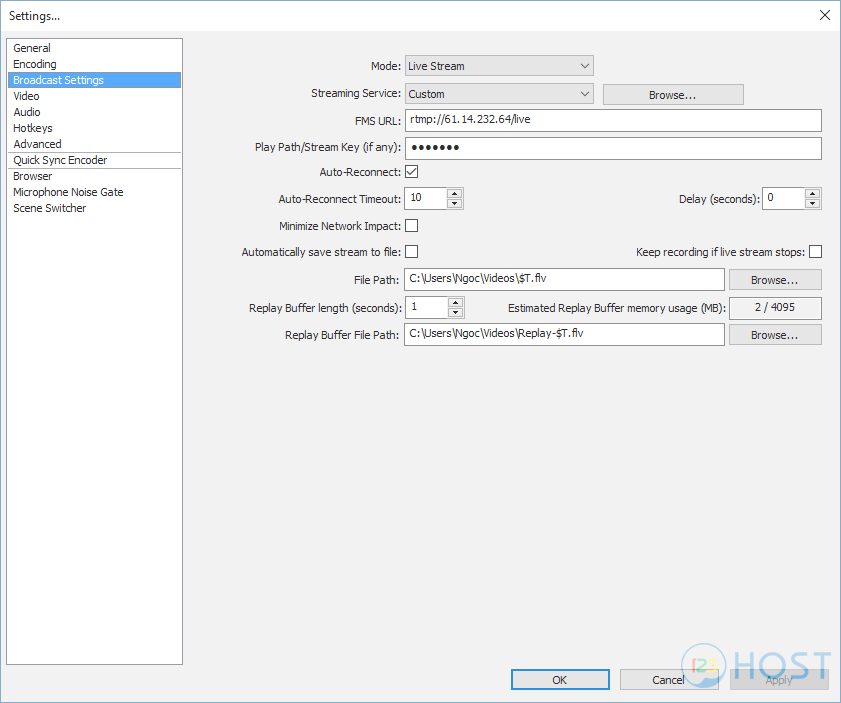
4. Dùng OpenBroadcaster để publish live video

– Tải về OpenBroadcaster tại địa chỉ <https://obsproject.com/>.

– Sau khi tải về và tiến hành cài đặt. Mở OpenBroadcaster lên giao diện như sau:



– Chọn Setting để cài đặt. Chọn tab Broadcast Settings:

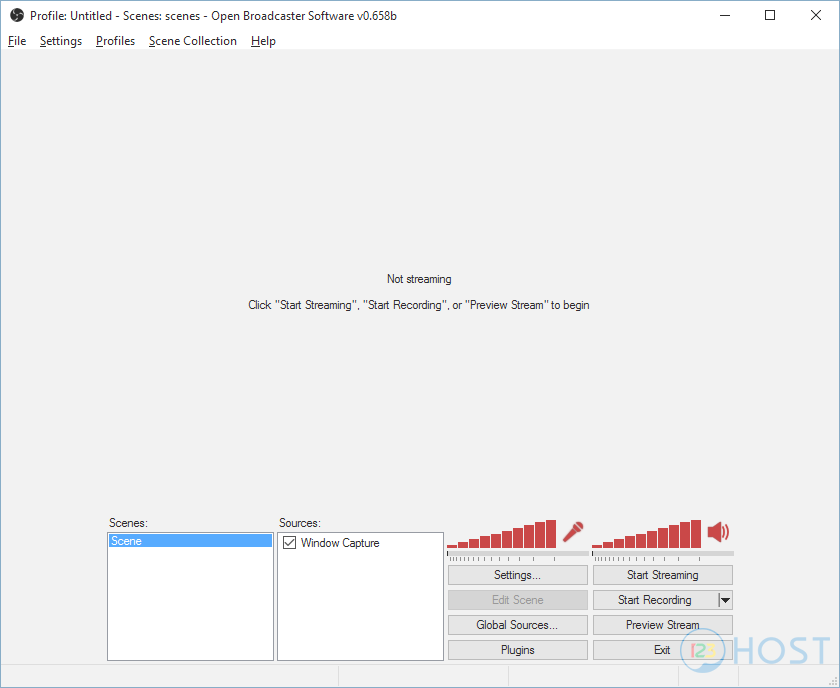


– Điền thông tin vào như trong ảnh:

            + **URL**: rtpm://61.14.232.64/live

            + **Stream Key**: 123host (điền gì cũng được)

– Quay lại màn hính chính ứng dụng trong khung Scenes tiến hàng tạo một màn hình bằng chuột phải lên đó chọn Add Screne và đặt tên cho nó:



– Tiếp theo trong khung **Sources**, chuột phải lên đó và chọn Add để thêm nguồn quay:

            + Quay của sổ

            + Quay màn hình

            + …

– Sau đó nhấn **Start Streaming** để bắt đầu stream:

