How can I stream a rtsp ip camera to a website via nginx

Asked 6 months ago Modified 6 months ago Viewed 968 times

0

I try to stream my RTSP-IP-Camera on a website. I use the Nginx webserver. My source in the html-code is:

```
<source src=rtmp://ip-address:1935/live/ type="application/x-mpegURL" />
```

To convert the rtsp stream i use this ffmpeg code:

```
ffmpeg -rtsp_transport tcp -i rtsp://user:password@ip-camera:554/h264Preview_01_main -
vcodec copy -acodec copy -f mp4 -y rtmp://ip-address:1935/live/
```

I get the error message "muxer does not support non seekable output Could not write header for output file #0 (incorrect codec parameters ?): Invalid argument"

I also tried to convert the rtsp-camera into a mp4 file and then access the file as source in my html code, but i couldnt read the file while ffmpeg wrote in it.

If you need more information let me know.

Thank you and have a nice day.

html nginx ffmpeg rtsp rtmp

Share Improve this question Follow

edited Nov 22, 2021 at 13:52

asked Nov 22, 2021 at 9:27

Vepok
13 2

I got it working with an installation of obs-studio. So the rtsp-ip-camera is as a vlc-source in obs. The obs sends the stream to my nginx rtmp server and in my html-code the rtmp-server is the source. My goal is to keep of obs-studio and send the camera stream directly as source in the html-code or to the rtmp-server application. The ip-camera is not able to send via rtmp. – Vepok Nov 22, 2021 at 13:56

I tried some more, but i wont reach my goal. Here is what i tried today: `<source src=/etc/nginx/html/test.m3u8 type="application/x-mpegURL" />` is the source in my html config. And my ffmpeg: `ffmpeg -v info -rtsp_transport tcp -i rtsp://user:password@cameraip:port/h264Preview_01_main -c:v copy -c:a copy -maxrate 400k -bufsize 1835k -pix_fmt yuv420p -flags -global_header -hls_time 10 -hls_list_size 6 -hls_wrap 10 -start_number 1 /etc/nginx/html/test.m3u8` – Vepok Nov 23, 2021 at 17:01

1 Answer

Sorted by: Highest score (default)

0

First, you should use format `flv` , not `mp4` . More, you should specify a stream(StreamKey for obs) like `livestream` :

```
ffmpeg -rtsp_transport tcp -i rtsp://user:password@ip-camera:554/h264Preview_01_main -c
copy \
  -f flv -y rtmp://ip-address:1935/live/livestream
```

Then you covert RTSP to RTMP, and you can use server to covert the RTMP to HLS, like what you did.

The latency of HLS is large, about 5~10s, if you want to get lower latency, please use HTTP-FLV or WebRTC, see link here

Share Improve this answer Follow

answered Dec 18, 2021 at 10:49

Winlin
706 2 17

Thanks for your response. It worked. Now i just need to find out how to scale the stream. – Vepok Apr 28 at 8:38

Good news! Well I don't understand what's *scale the stream*, I think it's better to discuss on discord first. You could find the discord of srs in github.com/ossrs/srs – Winlin May 1 at 0:36