2-SRS 4.0配置支持WebRTC推拉流

课程链接： https://ke.qq.com/course/468797?tuin=137bb271

网页版本：https://www.yuque.com/docs/share/3415d552-e619-4c95-94b5-9182c3dd90f7?#《2-SRS 4.0配置支持WebRTC推拉流》

0 SRS 4.0流媒体服务器入门系列

结合SRS官方Wiki以及本人对SRS的理解，推出《SRS 4.0流媒体服务器入门系列》，包括内容：

1. SRS 4.0 开发环境搭建
2. SRS 4.0 配置支持WebRTC推拉流
3. SRS 4.0 RTMP推拉流转发原理，包括延迟分析
4. SRS 4.0 支持WebRTC一对一通话，包括信令原理讲解(待录制)
5. SRS 4.0 支持WebRTC多人通话，包括信令原理讲解(待录制)

6. SRS 4.0 RTMP to WebRTC原理分析(待录制)
7. SRS 4.0 WebRTC to RTMP 原理分析(待录制)
8. SRS 4.0 配置支持GB28181推流(待录制)
9. .....其他待规划

每篇文章配合对应的视频，本文档配套视频：2-SRS 4.0 配置支持WebRTC推拉流，大家可以扫码关注“零声学院”公众号，获取后续的更新。

零声学院

云服务器：阿里云Ubuntu 16.04

服务器：SRS(Simple Realtime Server，支持RTMP、HTTP-FLV、HLS、WebRTC)
推流端：ffmpeg + OBS
拉流端：ffplay +VLC + srs播放器

SRS官网：http://www.ossrs.net/
微信官方公众号 srs-server

SRS开源服务器

微信扫描二维码，关注我的公众号

1 启动支持rtc的srs服务

webrtc是默认支持的（--rtc=on），所以我们不需要重新编译：

我们直接使用默认的rtc配置(conf/rtc2rtmp.conf)跑起来：

cd srs.4.0/trunk**conf**/rtc2rtmp.**conf**

默认rtc2rtmp.conf配置如下

```
listen              1935;
max_connections     1000;
daemon              off;
srs_log_tank        console;

http_server {
    enabled         on;
    listen          8080;
    dir             ./objs/nginx/html;
}

http_api {
    enabled         on;
    listen          1985;
}
stats {
    network         0;
}
rtc_server {
    enabled on;
    # Listen at udp://8000
    listen 8000;
    #
    # The $CANDIDATE means fetch from env, if not configed, use * as default.
    #
    # The * means retrieving server IP automatically, from all network interfaces,
    # @see https://github.com/ossrs/srs/wiki/v4_CN_RTCWiki#config-candidate
     #拉取流地址:使用本机地址或如下配置
```

```
    #candidate        $CANDIDATE;
    candidate 114.215.169.66; #设置为公网的地址
}

vhost __defaultVhost__ {
    rtc {
        enabled     on;
        rtc_to_rtmp on; # 支持RTC推流，RTMP拉流
    }
    http_remux {
        enabled     on;
        mount       [vhost]/[app]/[stream].flv;
    }
}
```

一定要把对应的端口服务打开。

异常处理 局域网webrtc可以拉流,正式环境黑屏

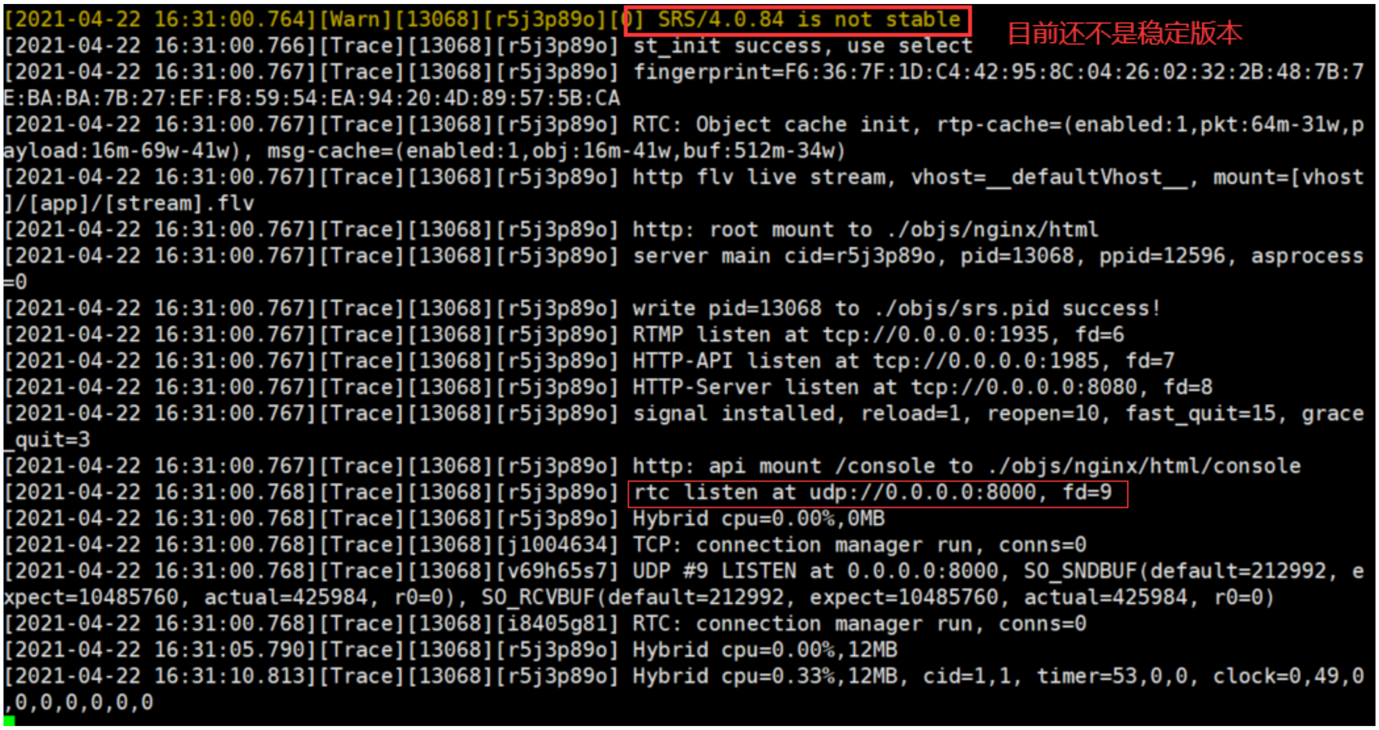
原因

```
rtc_server {
    enabled         on;
    listen          8000;
    candidate       $CANDIDATE;
}
```

rtc的$CANDIDATE在云服务器一般默认会选中172（之类）的局域网络，若是推流地址和拉流地址不在一个局域中会异常，解决方法设置为公网ip

```
rtc_server {
    enabled         on;
    listen          8000;
    candidate       114.215.xxx.xxx;# 设置为公网的地址
}
```

启动后，可以看到rtc监听的端口信息



2 WebRTC拉流演示

我们通过RTMP进行推流，然后通过WebRTC进行拉流。

2.1 推送RTMP视频

这里采用ffmpeg命令进行推流

```
ffmpeg -re -i  time.flv  -vcodec copy -acodec copy  -f flv -y rtmp://114.215.169.66/live/livestream
```

2.2 WebRTC拉流播放

推送流成功之后，使用srs自带的rtc_player播放器进行播放，直接请求srs服务的8080端口即可

http://114.215.169.66:8080/players/rtc_player.html

3 WebRTC推流演示

3.1 WebRTC推流

http://114.215.169.66:8080/players/rtc_publisher.html

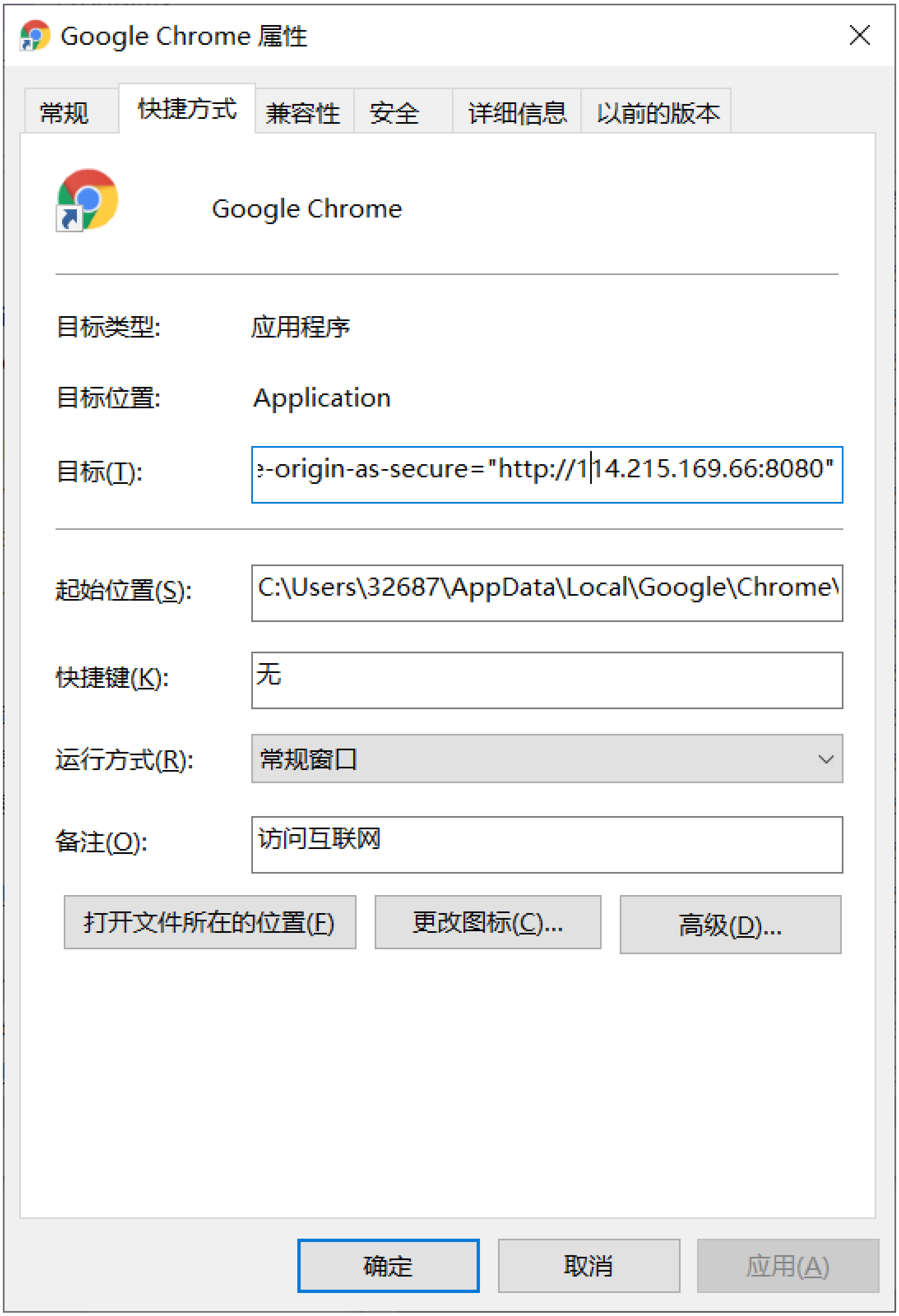
因为我们现在使用使用ip地址进行测试，没有使用https+域名的方式，所以在使用WebRTC时需要修改Chrome的启动参数。

在使用Chrome浏览器推流时会报：TypeError: Cannot read property 'getUserMedia' of undefined 错误，这个错误主要是https证书问题。

解决办法：先把chrome完全退出，右击桌面上chrome的快捷键，点击属性，在目标一栏添加如下内容，记着有个英文空格，IP换成自己的。
在目标加上以下参数（IP地址换成自己的）：

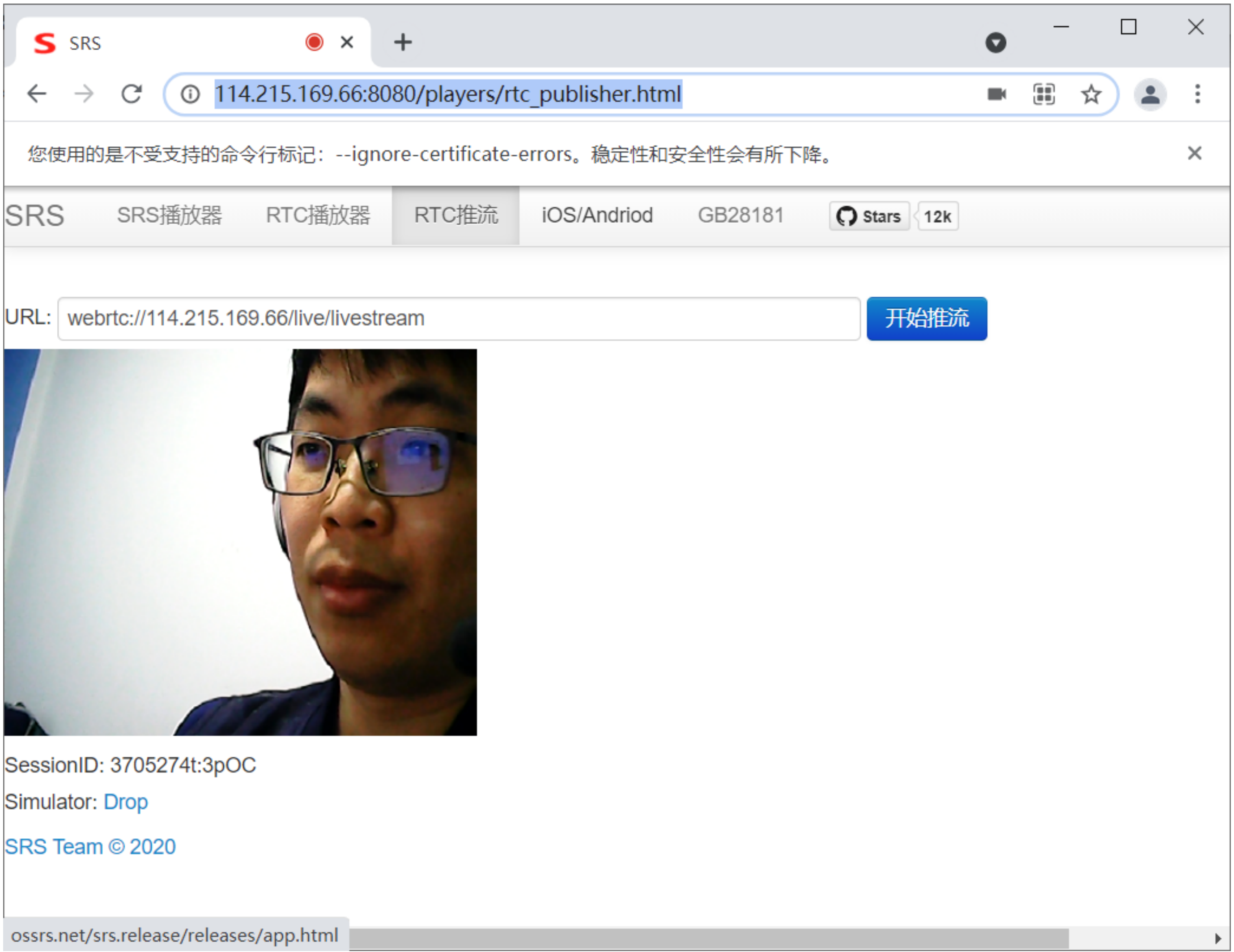
```
--ignore-certificate-errors --allow-running-insecure-content  --unsafely-treat-insecure-origin-as-secure="http://114.215.169.66:808
```

```
0"
```



然后重新打开chrome，输入自己的地址。

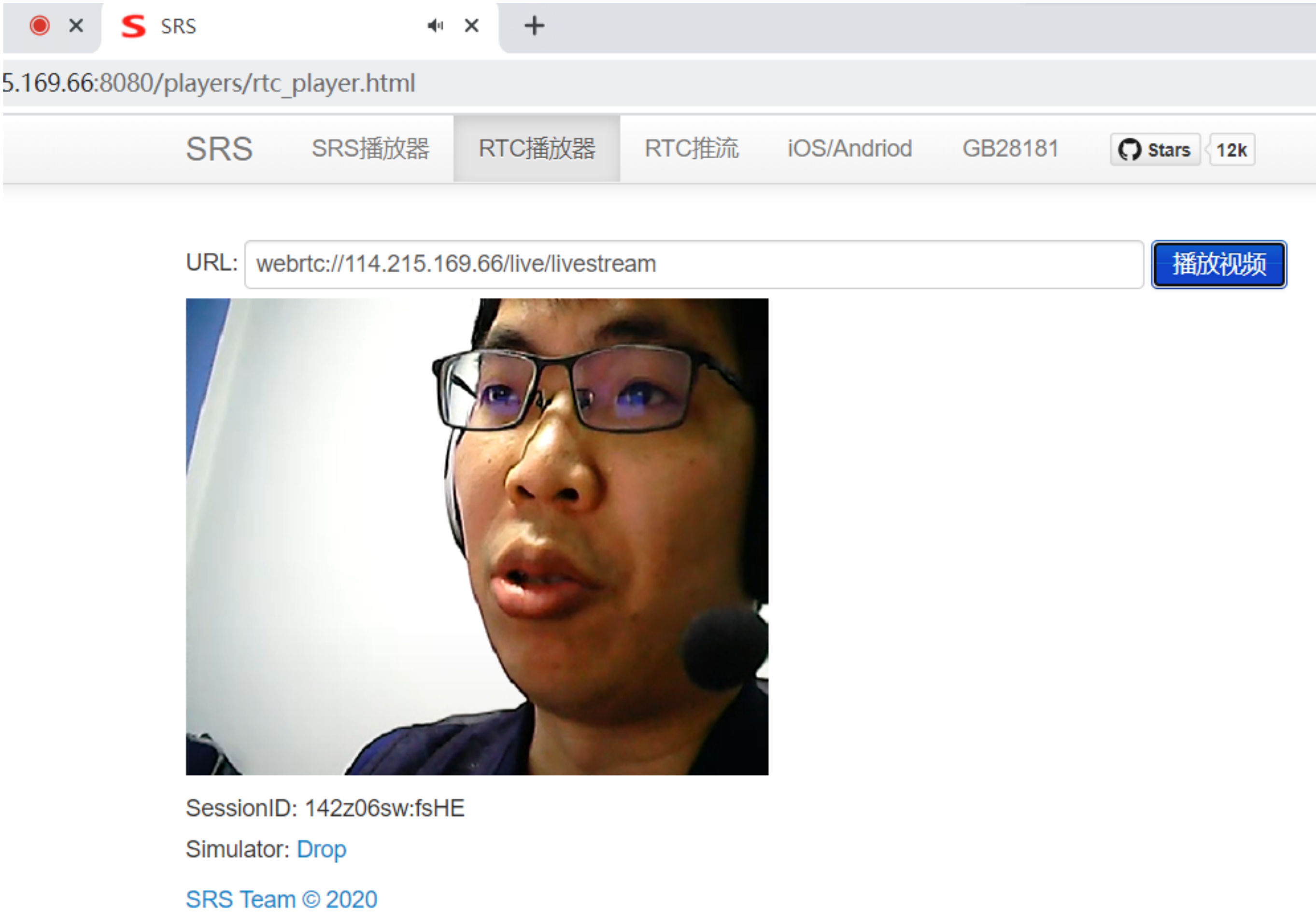
http://114.215.169.66:8080/players/rtc_publisher.html



需要使用http://114.215.169.66:8080/players/rtc_player.html去拉流。当前RTC推流还不支持RTMP/HTTP-FLV/HLS拉流。

3.2 WebRTC拉流播放

http://114.215.169.66:8080/players/rtc_player.html



3.3 ffplay和srs播放器拉流

ffplay rtmp://114.215.169.66/live/livestream

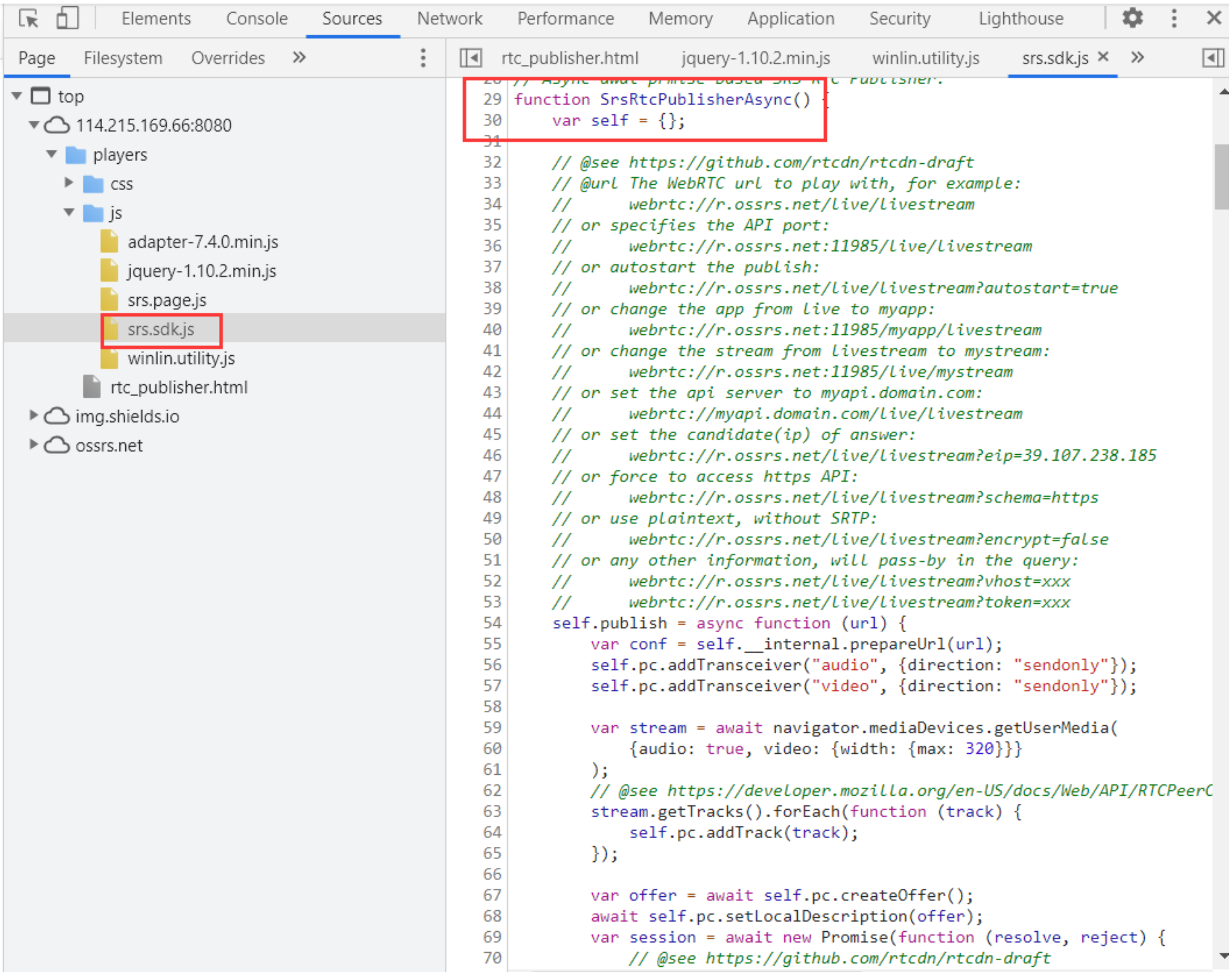
和RTMP推流时的播放是一样的。

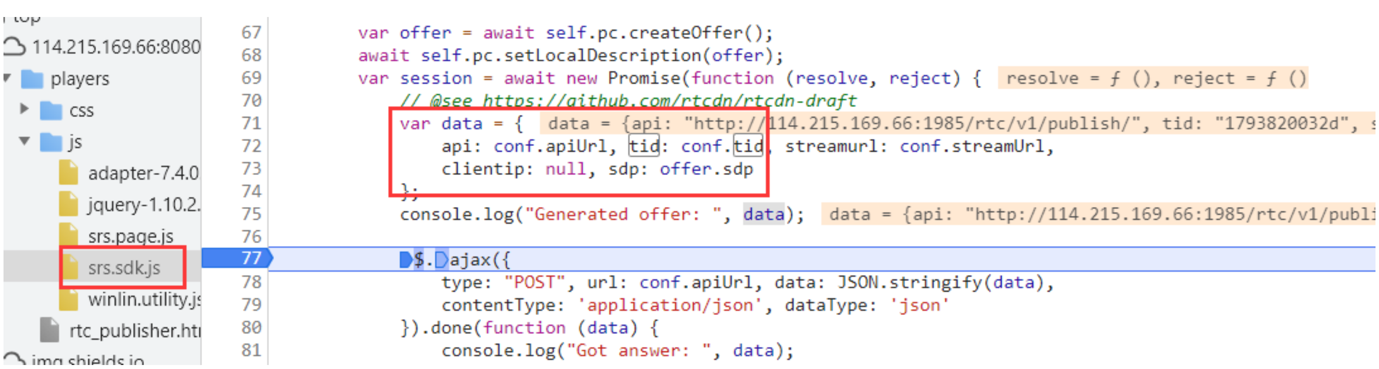
SRS播放器同理。

4 Web端的srs rtc推拉流sdk

对应的js文件为 srs.sdk.js

4.1 推流





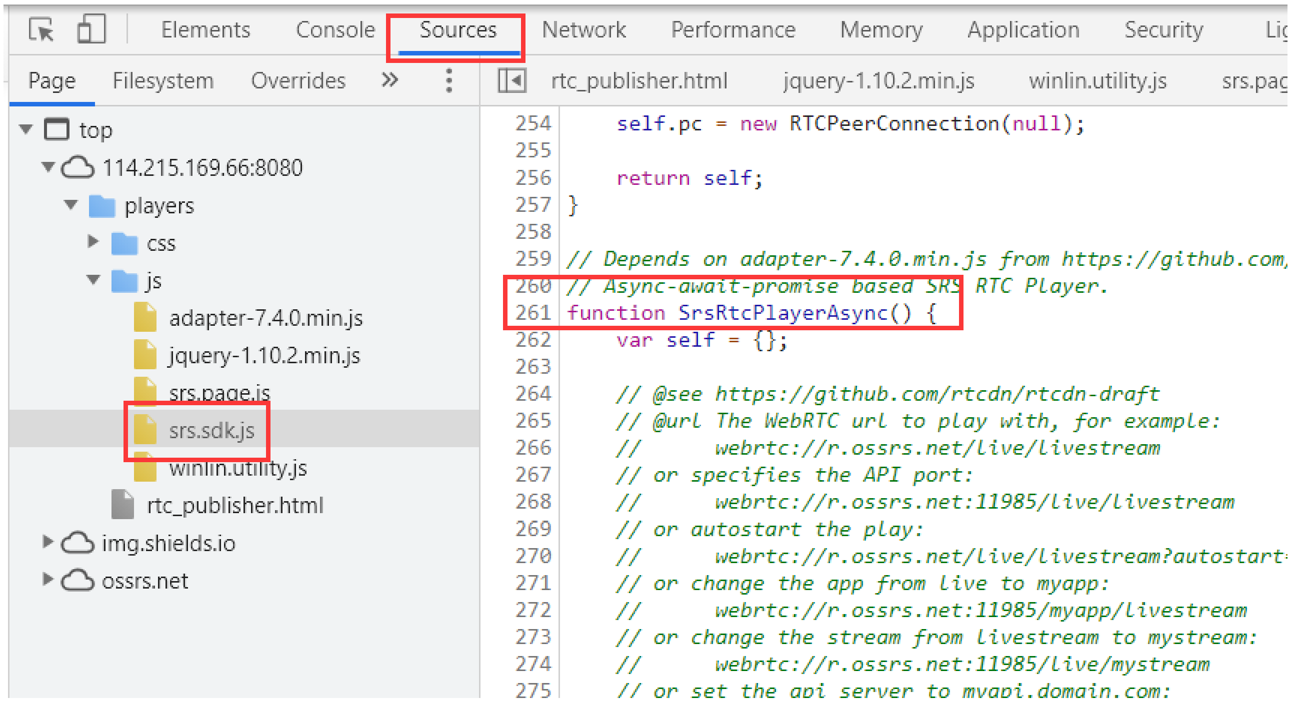
推流信令对应的URL：http://114.215.169.66:1985/rtc/v1/publish/

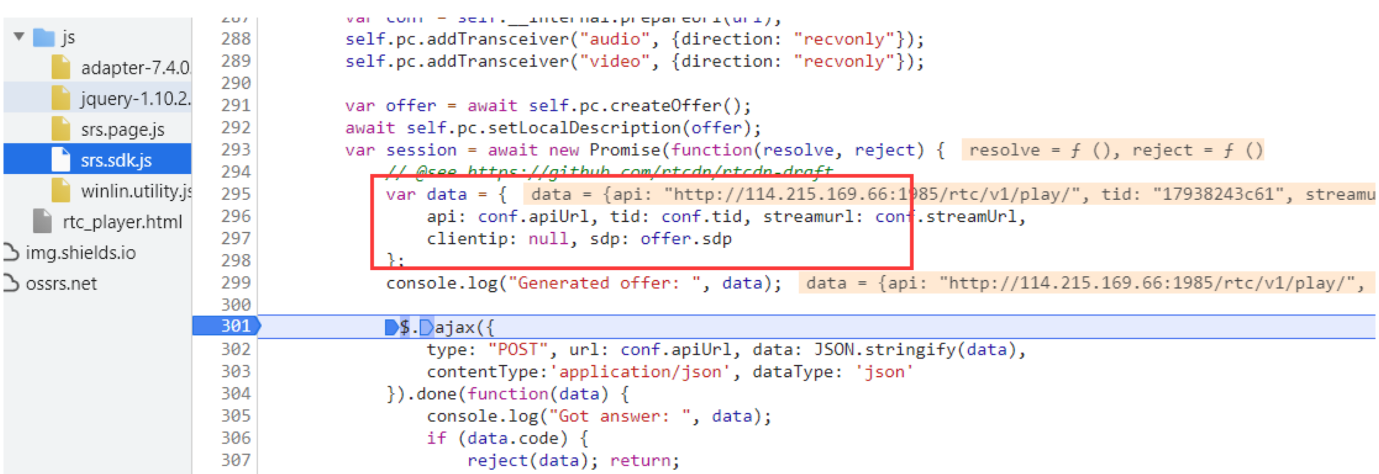
推流信令对应的data：

{   "api": "http://114.215.169.66:1985/rtc/v1/publish/",
    "tid": "1793820032d",
    "streamurl": "webrtc://114.215.169.66/live/livestream",
    "clientip": null,

"sdp": "v=0\no=- 170307602475242460 2 IN IP4 127.0.0.1\ns=-\nt=0 0\na=group:BUNDLE 0..\n"
}

4.2 拉流





拉流信令

{ "api": "http://114.215.169.66:1985/rtc/v1/play/",
"tid": "17938243c61",
"streamurl": "webrtc://114.215.169.66/live/livestream",
"clientip": null,
"sdp": "v=0\no=- 8914052225280550591 2 IN IP4 127.0.0.1\ns=-\nt=0 0\na=group:BUNDLE... \n"

}

4.2 更进一步阅读

更详细的candidate讲解请阅读官方Wiki：https://github.com/ossrs/srs/wiki/v4_CN_WebRTC