**前言：**

# 本篇文章是第一篇介绍SRS的文章，后续应该还会有一些测试和使用经验的文章，同时可能会有一些源码阅读笔记。本篇文章主要是介绍下国产开源流媒体SRS涉及的入门资料，SRS前期对标的nginx-rtmp-module，是国内第一款真正开源的流媒体服务，应用在直播和RTC领域。但是成立同学志存高远，不仅仅局限于某一个行业，近期WebRTC、SRT、GB28181支持的都很快，希望大家都能参与进来，支持国产开源流媒体发展壮大。

在我看来，推荐SRS作为流媒体人学习和使用的N个理由：

1. SRS最大的特点就是简单，表现在代码架构简单，实现简单，部署简单，运维简单；
2. 源代码彻底开源，采用了IMT非常宽松的开源协议，同时Git上有非常丰富的Wiki文档资料和提交了大量Issuse，非常适合学习和商用；
3. SRS有丰富的周边开发工具，有自研播放器和性能压测工具，分分钟搭建一套直播系统，直接可以上手相应环境验证功能和代码调试；
4. SRS已经迭代到4.0版本，功能很丰富，支持多源输入也支持多协议分发，你能看到的RTMP、HLS、DASH、RTSP、GB28181、WebRTC、SRT、HTTP-FLV都支持接入和分发；
5. SRS即支持小规模集群也支持大规模集群，特别是对CDN业务的关键特性的支持，性能稳定性都线上跑过和压测过，质量有一定的保障，定位成运营级的互联网直播服务器此言非虚；

**SRS官网介绍：**

SRS定位是运营级的互联网直播服务器集群，追求更好的概念完整性和最简单实现的代码。SRS提供了丰富的接入方案将RTMP流接入SRS， 包括[推送RTMP到SRS](https://github.com/ossrs/srs/wiki/v1_CN_SampleRTMP)、[推送RTSP/UDP/FLV到SRS](https://github.com/ossrs/srs/wiki/v2_CN_Streamer)、[拉取流到SRS](https://github.com/ossrs/srs/wiki/v1_CN_Ingest)。 SRS还支持将接入的RTMP流进行各种变换，譬如[将RTMP流转码](https://github.com/ossrs/srs/wiki/v1_CN_SampleFFMPEG)、[流截图](https://github.com/ossrs/srs/wiki/v3_CN_Snapshot)、 [转发给其他服务器](https://github.com/ossrs/srs/wiki/v3_CN_SampleForward)、[转封装成HTTP-FLV流](https://github.com/ossrs/srs/wiki/v3_CN_SampleHttpFlv)、[转封装成HLS](https://github.com/ossrs/srs/wiki/v3_CN_SampleHLS)、 [转封装成HDS](https://github.com/ossrs/srs/wiki/v2_CN_DeliveryHDS)、[支持SRT流](https://github.com/ossrs/srs/wiki/v4_CN_SRTWiki)、[录制成FLV/MP4](https://github.com/ossrs/srs/wiki/v3_CN_DVR)。SRS包含支持大规模集群如CDN业务的关键特性， 譬如[RTMP多级集群](https://github.com/ossrs/srs/wiki/v3_CN_SampleRTMPCluster)、[源站集群](https://github.com/ossrs/srs/wiki/v3_CN_OriginCluster)、[VHOST虚拟服务器](https://github.com/ossrs/srs/wiki/v3_CN_RtmpUrlVhost)、 [无中断服务Reload](https://github.com/ossrs/srs/wiki/v1_CN_Reload)、[HTTP-FLV集群](https://github.com/ossrs/srs/wiki/v3_CN_SampleHttpFlvCluster)。此外，SRS还提供丰富的应用接口， 包括[HTTP回调](https://github.com/ossrs/srs/wiki/v3_CN_HTTPCallback)、[安全策略Security](https://github.com/ossrs/srs/wiki/v2_CN_Security)、[HTTP API接口](https://github.com/ossrs/srs/wiki/v3_CN_HTTPApi)、 [RTMP测速](https://github.com/ossrs/srs/wiki/v1_CN_BandwidthTestTool)。SRS在源站和CDN集群中都得到了广泛的[应用Applications](https://github.com/ossrs/srs/wiki/v1_CN_Sample)。

**SRS基本资料：**

Wiki地址，里面有上百篇详细文档，第一入手资料：

<https://github.com/ossrs/srs/wiki>

Git分支，可以看源码：

<https://github.com/winlinvip/srs>

官网，有安装包，客户端供大家测试体验：

<http://www.ossrs.net/srs.release/releases/>

微信公众号，发布SRS最新动态：



SRS还有微信讨论群，可以加我微信拉大家进去学习讨论：



**流媒体参考资料：**

参考资料1，包含了SRS3.0源码阅读笔记，也是SRS支持国标的开发者。同时分支不局限SRS学习，还有大量RTC领域的资料，还包括cdn原理实现，连麦，会议直播录相，微信小程序会议接入，sip客户端会议接入等方案文档，物联网调查报告等：

<https://github.com/xialixin/srs_code_note>

参考资料2，我的git项目，主要包含各种流媒体涉及的编码、封装、传输协议等文档的资料整理，以及一些测试素材和测试工具：

<https://github.com/ty6815/AvStackDocs>

**支持监控GB28181背景和定位：**

SRS是一款开源流媒体，适合直播以及低延时流媒体领域，但是从来不设限，可以应用在直播、视频会议、在线教育等场景。同时视频领域还有两大块、广电和视频监控，随着互联网的迅速发展，视频监控设备上云的需求越来越多，以前还可以私有化局域网里面搞一搞，但是随着移动互联网的发展，视频监控有大量的移动端直播需求，让人随时随地能看处理监控报警事件、查看回放视频是刚需。同时这几年红黄蓝幼儿园事件、不良商家黑暗料理问题层出不穷，所以视频监控从行业逐渐走向民用，平安校园、明厨亮灶、智慧社区、智能家居等场景的落地，更是加快了视频监控上云的趋势。

SRS我看Issue大概是19年有人提问，然后19年后半年成立这边应该做过一些调研，SRS也跟进了下，但是跟进方向不对，年末跟成立聊过这个问题，才最终确定SRS开始支持这块。

## 关于视频监控上云前面写过文章，分析过《视频监控摄像头的互联网化实践思路》，简单来说：

1. SRS如果支持RTSP拉流，由于摄像头一般都在局域网，没有外网访问途径，如果要把SRS作为代理服务部署在用户侧，这用起来不靠谱也不符合SRS的定位，pass掉了，也意味着ONVIF协议集成到SRS里面没有多大意义；
2. 既然主动拉流不靠谱，那就支持摄像头推流，SRS目前支持RTSP推流，但是摄像头目前支持该种方法的很少，一般都是在局域网拉流，所以SRS支持了没用，设备端不支持；
3. 其次部分设备端支持RTMP推流，那么这点作为SRS基本功能肯定是支持的，但是并不是所有设备端都支持RTMP推流，设备也不仅仅包含摄像头，还包括NVR，边缘服务器等；
4. 推流既然是方向，还有没有其它形式支持设备端推流的，有的，国内设备端用的国家标准GB28181协议，除了消费类家庭摄像头，基本大部分摄像头都支持该协议，便宜的二三百，贵点的两三千设备，都是支持这个协议，所以SRS只有支持GB28181，就能覆盖90%以上的视频监控上云需求，其次支持摄像头、NVR、下级国标平台性软件统统上云；

所谓的国标GB28181协议大家简单理解为SIP+RTP即可，所以SRS为了支持视频监控行业设备上云，还是要支持国标GB28181协议。

但是支持GB28181协议，如果把信令和媒体收在SRS里面实现，违反了SRS的简单原则，其次信令这块对接起来非常麻烦，虽然是国家标准，但是过个厂家实现的不标准，有很多细小问题需要接入方来兼容，其次这块还有大量业务在里面，所以SRS不应该去完整实现GB协议信令部分，只需要实现媒体部分即可，换句话说只要能摄像头接进来，能把媒体流从RTP转分发为RTMP、HLS等即可。信令部分只需要提供基本的测试接入部分即可，目的只是为了SRS能跑起来，信令部分还是需要使用SRS的同学们自己实现，媒体部分会提供RestFul接口供信令调用。

**实际测试情况：**

**国标部分编译和安装**