9-SRS4.0-SRTP加密传输

libsrtp 的使用

零声学院：音视频高级课程：https://ke.qq.com/course/468797?tuin=137bb271

本文版权归腾讯课堂 零声教育所有，侵权必究。

重点知道采用srtp库实现加解密，原理可以参考《WebRTC 传输安全机制第二话：深入显出 SRTP 协议》https://www.cnblogs.com/9849aa/p/14889687.html

libsrtp 的使用

libsrtp 是被广泛使用的 SRTP/SRTCP 加密的开源项目。经常用到的 api 如下：

1. srtp_init，初始化 srtp 库，初始化内部加密算法，在使用 srtp 前，必须要调用了。
2. srtp_create，创建 srtp_session，可以结合本文中介绍的 session，session key 等概念一起理解。
3. srtp_unprotect/srtp_protect，RTP 包加解密接口。
4. srtp_protect_rtcp/srtp_unprotect_rtcp，RTCP 包的加解密接口。
5. 5.srtp_set_stream_roc/srtp_get_stream_roc，设置和获取 stream 的 ROC，这两个接口在最新的 2.3 版本加入。

重要的结构 srtp_policy_t，用来初始化加解密参数，在 srtp_create 中使用这个结构。以下参数需要关注：

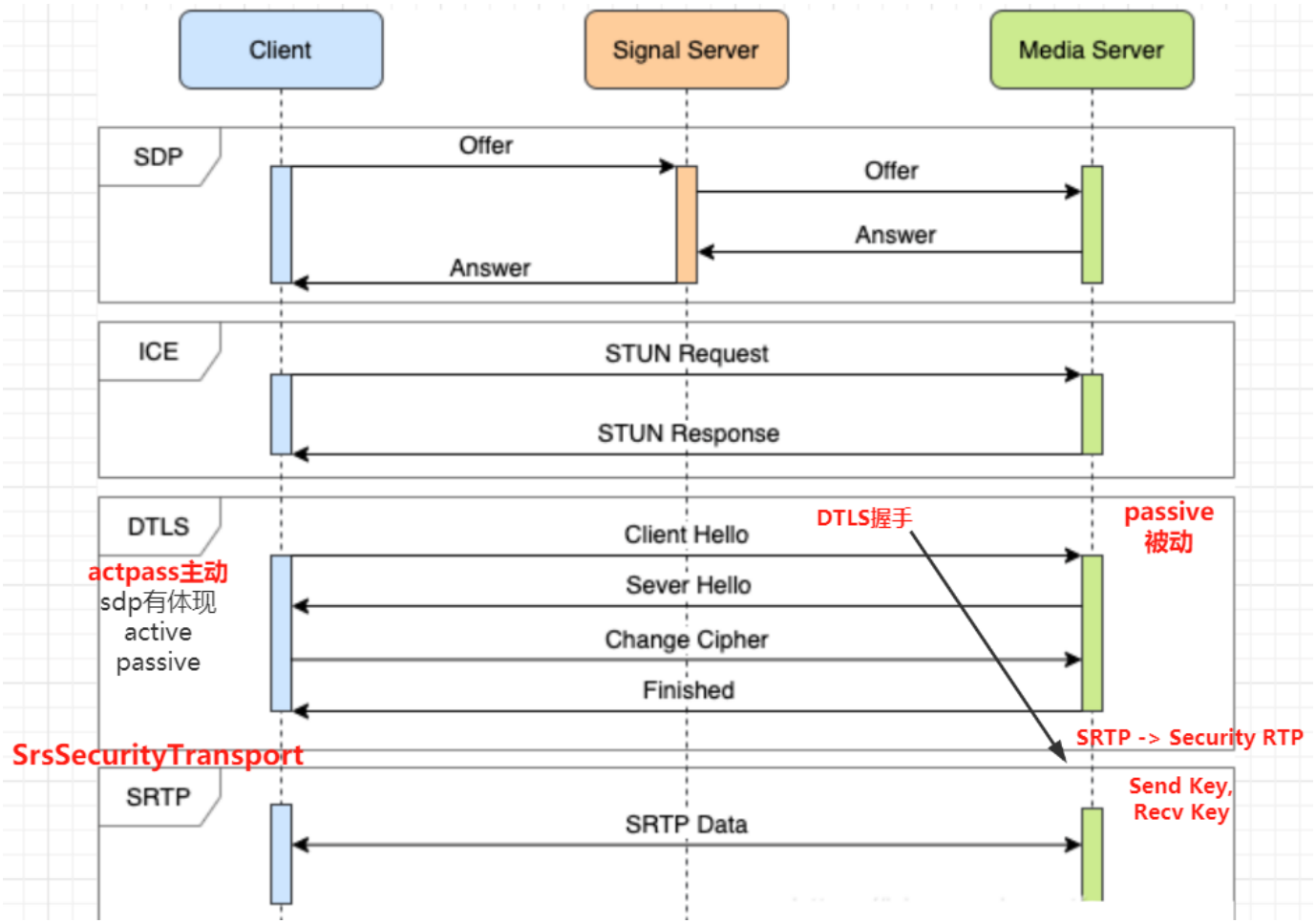
1. DTLS 协商后得到的 MasterKey 和 MasterSalt 通过这个结构传递给 libsrtp，用于 session key 的生成。
2. window_size，对应我们之前描述的 srtp 防重放攻击的窗口大小。
3. allow_repeat_tx，是否允许重传相同序号的包。

原理可以参考：《WebRTC 传输安全机制第二话：深入显出 SRTP 协议》
https://www.cnblogs.com/9849aa/p/14889687.html

SRTP的内容比较简单

```
class SrsSRTP{
private:
    srtp_t recv_ctx_;  // 用于解密
    srtp_t send_ctx_;  // 用于加密
public:
    SrsSRTP();
    virtual ~SrsSRTP();
```

```
public:
    // Intialize srtp context with recv_key and send_key.
    srs_error_t initialize(std::string recv_key, std::string send_key);
public:
    srs_error_t protect_rtp(void* packet, int* nb_cipher); // 加密RTP数据
    srs_error_t protect_rtcp(void* packet, int* nb_cipher); //加密的是RTCP数据
    srs_error_t unprotect_rtp(void* packet, int* nb_plaintext); //解密RTP数据
    srs_error_t unprotect_rtcp(void* packet, int* nb_plaintext);//解密RTCP数据
};
```



加密

Breakpoint 3, SrsSRTP::protect_rtp (this=0x5555562846c0, packet=0x5555563846f0, nb_cipher=0x55555621ddb4) at src/app/srs_app_rtc_dtls.cpp:1118

1118 srs_error_t err = srs_success;

(gdb) bt

#0 SrsSRTP::protect_rtp (this=0x5555562846c0, packet=0x5555563846f0,
nb_cipher=0x55555621ddb4)
    at src/app/srs_app_rtc_dtls.cpp:1118
#1 0x00005555557daa13 in SrsSecurityTransport::protect_rtp (this=0x555556513100,
packet=0x5555563846f0,
    nb_cipher=0x55555621ddb4) at src/app/srs_app_rtc_conn.cpp:196
#2 0x00005555557e7f14 in SrsRtcConnection::do_send_packet (this=0x55555637c970,
pkt=0x55555634e0f0)
    at src/app/srs_app_rtc_conn.cpp:2552
#3 0x000055555582af41 in SrsRtcAudioSendTrack::on_rtp (this=0x5555562c93b0,
pkt=0x55555634e0f0)
    at src/app/srs_app_rtc_source.cpp:2591
#4 0x00005555557dd8c1 in SrsRtcPlayStream::send_packet (this=0x55555637e4a0,
pkt=@0x55555621def8: 0x55555634e0f0)
    at src/app/srs_app_rtc_conn.cpp:673
#5 0x00005555557dd2a7 in SrsRtcPlayStream::cycle (this=0x55555637e4a0) at
src/app/srs_app_rtc_conn.cpp:608
#6 0x000055555571380a in SrsFastCoroutine::cycle (this=0x55555622b880) at
src/app/srs_app_st.cpp:270
#7 0x00005555557138a6 in SrsFastCoroutine::pfn (arg=0x55555622b880) at
src/app/srs_app_st.cpp:285
#8 0x000055555583bdfa in _st_thread_main () at sched.c:363
#9 0x000055555583c696 in st_thread_create (start=0x100000000, arg=0x0, joinable=21845,
stk_size=1447353216) at sched.c:694
#10 0x0000000000000000 in ?? ()