Популярно про SRT

В мире продолжается бурное развитие сети Интернет, как в части охвата территории, так и в части увеличения пропускной способности сети. Уже сегодня сеть Интернет эффективно конкурирует с традиционными средствами доставки телесигналов – спутниковыми, кабельными и эфирными. Однако сеть Интернет имеет свою специфику – это возможные потери пакетов данных, разные задержки пакетов, ограничения трафика. Все это усложняет использование такой сети для передачи потоковых сервисов. Чтобы компенсировать эти особенности потребовалась буферизация сигналов, а также разработка специальных протоколов передачи с возможностью восстановления потерянных пакетов и коррекцией задержек передачи.

Традиционный протокол сети Интернет TCP/IP, в принципе, может использоваться для передачи видеосигналов, однако он имеет непредсказуемые задержки, значительный избыточный трафик и, в некоторых ситуациях, может привести к блокировкам сервиса. Поэтому он практически не применяется для передачи потоковых сервисов.

Другой популярный протокол – UDP. Он значительно быстрее и проще в использовании, не создает избыточного трафика, но он не имеет средств коррекции ошибок. Поэтому в чистом виде он применяется только на сетях с гарантированным качеством сервиса. В результате его практическое использование ограничивается использованием для мультикастового вещания в частных сетях операторов передачи данных.

Относительно недавно для трансляции телевизионных сигналов по «публичным» сетям передачи данных стали использовать технологию OTT (подробнее об этом в «Теле-Спутник» № 6 (272) / июнь 2018). Эта технология широко используется для доставки телесигналов до индивидуальных абонентов. Однако эта технология плохо приспособлена для использования ее для профессиональной передачи телесигналов. Она имеет значительную и непредсказуемую задержку сигнала, не имеет средств для защиты передаваемого контента и пригодна для передачи только SPTS видеосервисов.

А профессиональной индустрии требовался протокол для профессиональной передачи телевизионных сервисов по сети Интернет. Это означает что он должен был отвечать следующим требованиям – работать на неуправляемых сетях передачи данных с потерями, не создавать значительного избыточного трафика, иметь встроенную защиту передаваемого контента, работать с любым типом передаваемых данных (SPTS/MPTS/Data), иметь минимальную и стабильную задержку сигналов, иметь открытый код. Для создания такого протокола в 2017 году была основана группа SRT Alliance, членами которой на сегодняшний день являются более 300 компаний, включая компанию WISI.

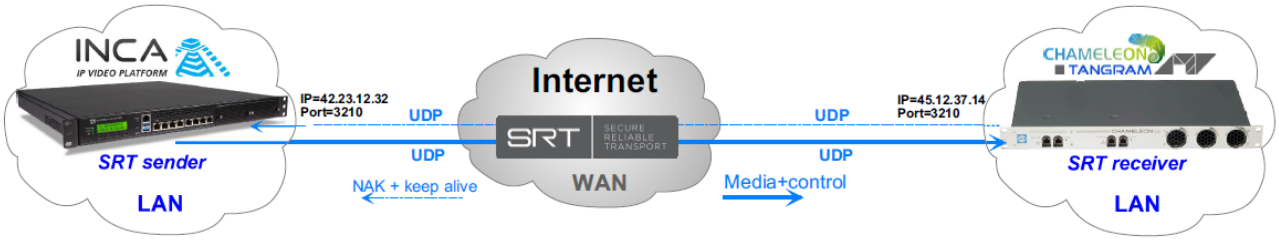
Результатом работы альянса стала разработка протокола передачи по «публичным» сетям Secure Reliable Transport - SRT. Этот протокол полностью отвечает поставленным требованиям. Нельзя сказать что у протокола SRT не было предшественников. На сегодняшний день для таких задач чаще всего используется протокол Real Time Messaging Protocol сокращённо англ. RTMP. Однако этот протокол по всем основным характеристикам проигрывает SRT. К тому же это проприетарный протокол компании Adobe Inc, что ограничивает круг совместимого оборудования и повышает стоимость решения.

В отличие от RTMP, SRT протокол является открытым и поддерживается широким ассортиментом оборудования от различных производителей.

При всех преимуществах не стоит забывать и об ограничениях протокола SRT - этот протокол хорошо приспособлен для профессиональной передачи «точка-точка», но не предназначен для многоадресной рассылки на массовых абонентов.

Как работает и как настроить передачу по SRT протоколу.

Как происходит передача данных через сеть Интернет по SRT протоколу, проиллюстрировано на диаграмме ниже.



Здесь в качестве передатчика (SRT sender) показано оборудование INCA, а в качестве приемника (SRT receiver) показано оборудование Chameleon/Tangram от фирмы WISI. Но на этих позициях можно использовать, также, и оборудование от любого другого производителя.

Передача данных через сеть производится по протоколу UDP. Это простой и быстрый протокол практически не создающий избыточного трафика.

А как же тезис о его ненадежности?

Для того чтобы обеспечить возможность коррекции ошибок, в пакеты передаваемых данных добавляется информация о номере пакета. При отсутствии потерь в передаче данных от передатчика к приемнику идет непрерывный поток данных по UDP протоколу. Дополнительный трафик не создается. Однако если приемное устройство обнаруживает что произошел пропуск номера пакета данных, то оно немедленно отсылает на передатчик сообщение о неприеме пакета (NAK). И передатчик повторно посылает на приемник потерянный пакет. Таким образом, дополнительный трафик появляется в сети только при потере данных и объем такого трафика прямо пропорционален интенсивности ошибок в такой сети.

Из описания процесса работы SRT протокола можно сделать два важных вывода, о задержке передачи и требуемой полосе пропускания канала.

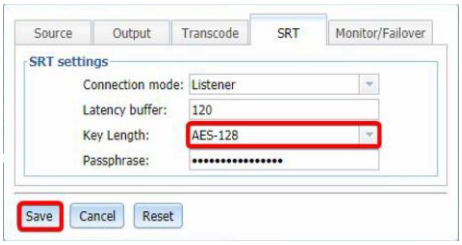
В параметрах состояния протокола вы увидите параметр RTT (round-trip time) – это время, затрачиваемое на прохождение сигнала от передатчика к приемнику и обратно. Этот параметр очень близок к привычному сетевикам параметру Ping. Очевидно, что для нормальной работы протокола SRT время буферизации (задержки) сигнала должно превышать значение RTT, так как для коррекции ошибок требуется время, гарантированно превышающее RTT. Типовой размер буфера выбирается при настройке в 3…6 раз больше, чем RTT и называется этот коэффициент - RTT мультипликатор. Чем меньше этот параметр, тем меньше задержка при передаче сигнала. Однако при уменьшении RTT мультипликатора вы оставляете меньше времени устройствам на коррекцию ошибок, и это требует наличия дополнительного запаса по пропускной способности вашего канала передачи. Этот требуемый запас зависит, также, от значения интенсивности потерь пакетов в сети передачи. Его можно оценить из следующей таблицы:

Потери пакетов не более (%)	RTT мультипликатор	Запас полосы пропускания (%)	Минимальная задержка SRT (для RTT ≤ 20 ms)
≤1	3	33	60
≤2	4	25	80
≤7	5	20	100
≤10	6	17	120

Размер буфера SRT настраивается пользователем по результатам контрольного измерения параметров линии связи (RTT или Ping). Размер буфера может быть установлен как на передатчике, так и на приемнике. В процессе связи будет автоматически использовано максимальное значение размера буфера приемника или передатчика.

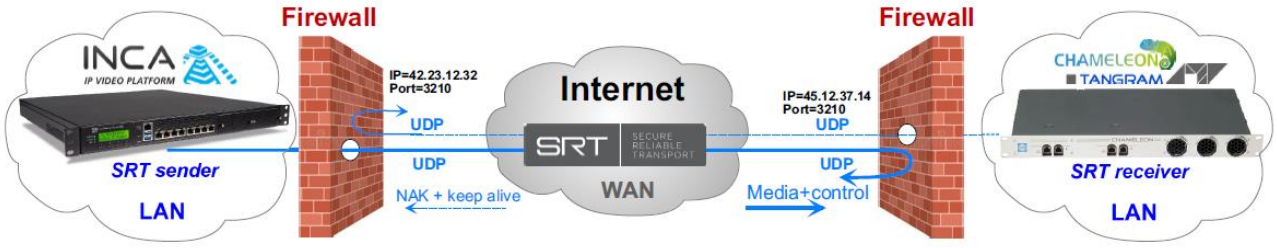
Кроме того, протокол SRT имеет встроенное AES скремблирование передаваемых данных. Чтобы заскремблировать передаваемые потоки достаточно в меню настройки SRT передатчика выбрать длину ключа (Key Length) AES-128 или AES-256 и задать пароль (Passphrase). Пароль может содержать от 10 до 79 UTF-8 печатных символов. Если пароль не задан, то скремблирование не производится.

Для приема заскремблированного SRT потока, на приемном устройстве достаточно задать тот же режим скремблирования и тот же пароль.



Гладко было на бумаге …

Однако в реальных ситуациях проблема с установлением соединения между передатчиком и приемником выглядит значительно сложнее. Как правило, между внутренней сетью клиентов (LAN) и сетью Интернет (WAN) всегда присутствуют защитные экраны – Firewall, которые блокируют внешние потоки, которые не запрошены клиентом или не входят в политики фильтрации контента. Клиент также может быть подключен через NAT.

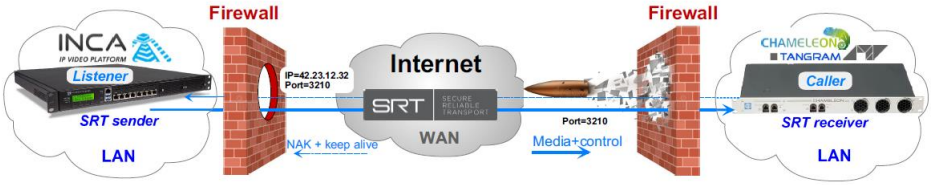


Файрволы могут находиться как под управлением клиента, так и провайдера Интернета, что существенно усложняет процесс установления соединения между SRT передатчиком (sender) и SRT приемником (receiver)

В протоколе SRT для обхода этих ограничений существуют несколько различных режимов установления соединений – Caller, Listener, Rendezvous.

Caller

Ниже показана ситуация, когда SRT передатчик установлен за Firewall с известным внешним IP и возможностью открыть нужные порты, а SRT приемник установлен за неконтролируемым пользователем Firewall.



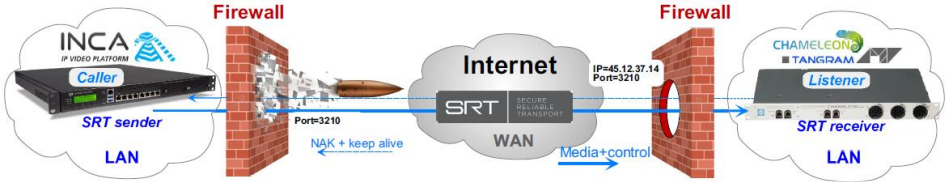
Инициатором обмена является запрос приемника, который временно открывает «окно» от WAN к приемнику по требуемому порту

В этом режиме SRT передатчик является пассивным «слушателем» (Listener), а SRT приемник «запросчиком» (Caller) – инициатором начала передачи.

В этом режиме SRT приемник может иметь динамический IP адрес с NAT.

Listener

Ниже показана обратная ситуация, когда SRT приемник установлен за Firewall с известным внешним IP и возможностью открыть нужные порты, а SRT передатчик установлен за неконтролируемым пользователем Firewall.

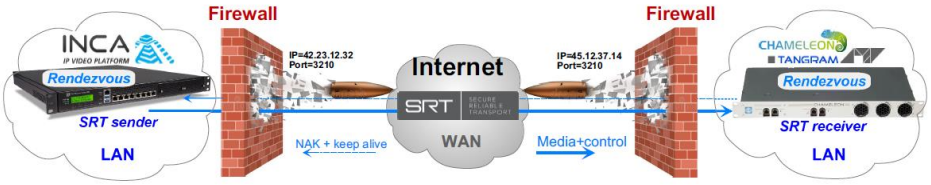


Инициатором обмена является SRT передатчик, который временно открывает «окно» от своей LAN к WAN по требуемому порту.

В этом режиме SRT передатчик является «запросчиком» (Caller) – инициатором начала передачи, а SRT приемник «слушателем» (Listener). В этом режиме SRT передатчик может иметь динамический IP адрес с NAT.

Rendezvous

Это самая сложная ситуация, когда оба, SRT приемник и передатчик, установлены за неконтролируемыми пользователями Firewall.



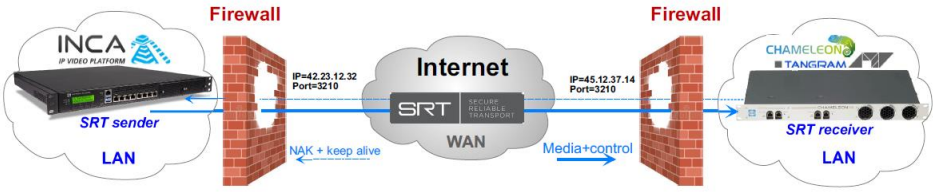
Инициаторами обмена являются одновременно оба, SRT передатчик и приемник, которые совместно временно открывает «окна» от своей LAN к WAN по требуемому порту.

В этом режиме оба SRT оконечные устройства являются инициаторами начала передачи. В этом режиме SRT передатчик и приемник должны иметь известные IP адреса и одинаковые порты.

Что после установления соединения?

После установления соединения, независимо от способа его установления, обмен между SRT передатчиком и приемником ведется одинаково для всех режимов

Открытые порты в Firewall поддерживаются в квазипостоянном состоянии за счет передачи медиаданных в прямом направлении и служебной информации в обратном направлении.



При отсутствии ошибок, когда обратные запросы по повторной посылке пакетов отсутствуют, каждые 10 мс передается служебный пакет «keep alive» для поддержания обратного соединения.

Протокол SRT подтвердил свою надежность на практике при освещении ответственного события - трансляции Парада Победы на Красной площади, посвященного 75-летию победы в Великой Отечественной войне (https://telesputnik.ru/materials/tech/article/istoriya-odnoy-translyatsii/).

Надеюсь, что приведенная информация поможет операторам корректно настроить передачу по протоколу SRT и эффективно использовать все его возможности.

Вячеслав Чулков,
Технический эксперт WISI.