SRT Protocol Overview

IETF 107 [dispatch]

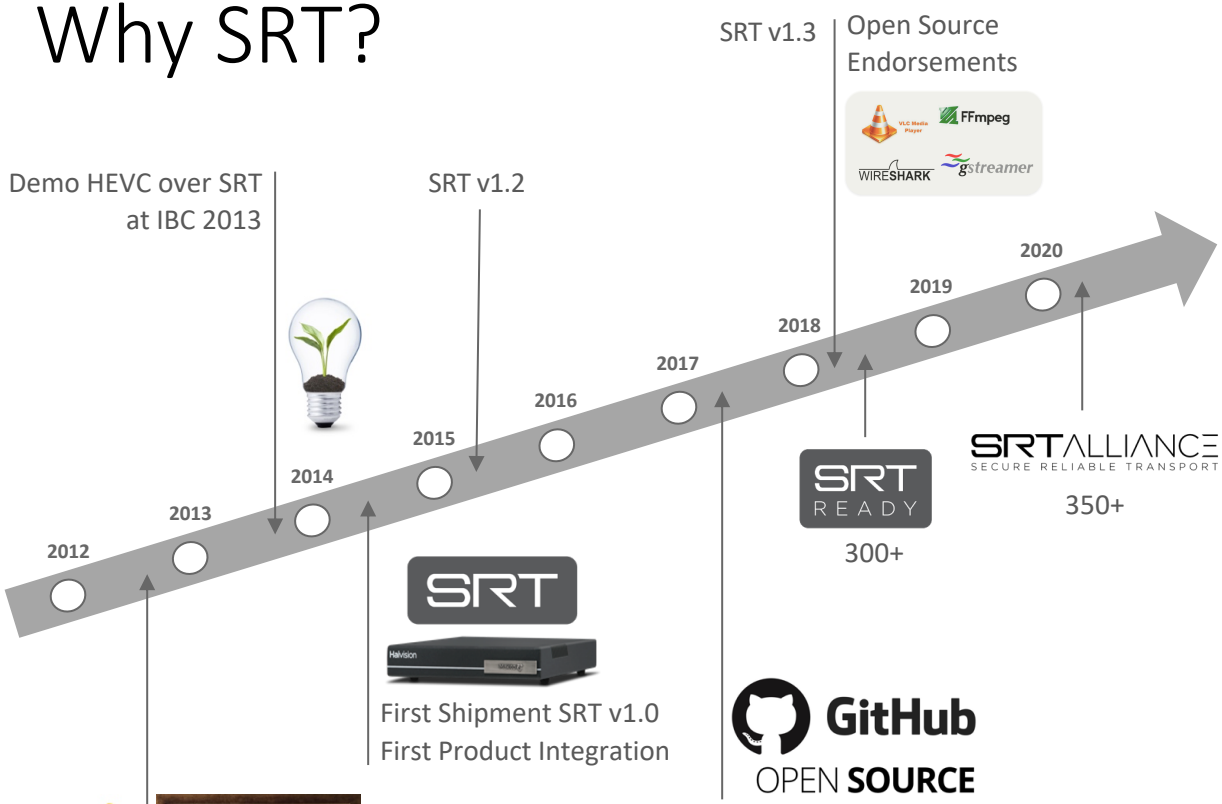
Maxim Sharabayko, PhD
Senior Software Developer | Haivision
maxsharabayko@haivision.com

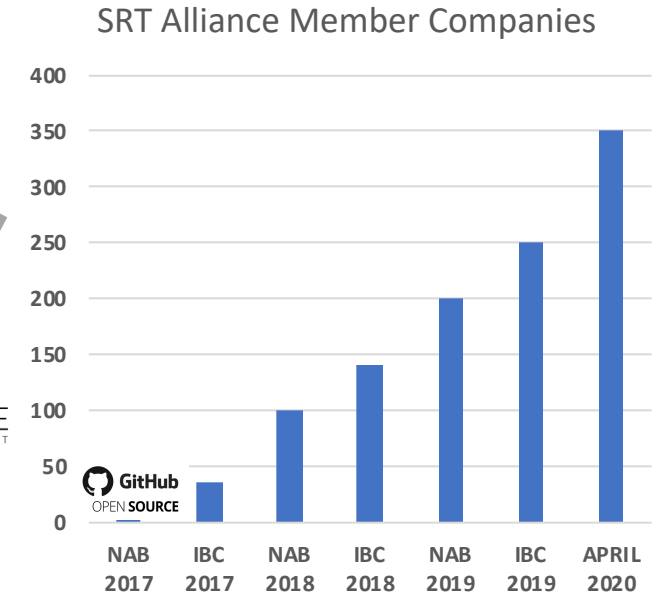
What is SRT?

- A protocol on top of UDP (unicast)
- Content agnostic
- Bidirectional data transfer
- ARQ (ACK + NAK)
- FEC (Packet Filter API v1.4.0+)
- Connection bonding (v1.5.0)
- Stream multiplexing
- Secure (AES 128/192/256 Encryption)

*Enabling **low-latency video** contribution & distribution and **fast file transfer** over unpredictable networks.*

Why SRT?


SRT v1.3
Open Source Endorsements
Demo HEVC over SRT at IBC 2013
SRT v1.2
2012
2013
2014
2015
2016
2017
2018
2019
2020
SRT READY
300+
SRTALLIANCE
SECURE RELIABLE TRANSPORT
350+
SRT
First Shipment SRT v1.0
First Product Integration
GitHub
OPEN SOURCE


SRT Alliance Member Companies
400
350
300
250
200
150
100
50
0
GitHub
OPEN SOURCE
NAB 2017
IBC 2017
NAB 2018
IBC 2018
NAB 2019
IBC 2019
APRIL 2020

Why IETF?

1. Implementation → official specification.
2. SRT + IETF → improve SRT.
3. Knowledge exchange.

SRT Operation Modes

Message Mode

- Non-real-time
- File/message transfer
- Content agnostic
- Message interchange

Live Mode

- Real-time
- Live Streaming
- Content agnostic
- Latency management

Buffer Mode

- Non-real-time
- A single file transfer
- Content agnostic

SRT Live Streaming

- Data Transmission Mode: “Live”
- A fixed end-to-end latency:
  - Network delay
  - Configurable receiver buffer delay
- Recovers source timing T_{SOURCE} : $T_{DELIVERY} >= T_{SOURCE} + T_{LATENCY}$
  - $T_{DELIVERY}$ – packet delivery time
  - T_{SOURCE} – packet source time
  - $T_{LATENCY}$ – end-to-end latency

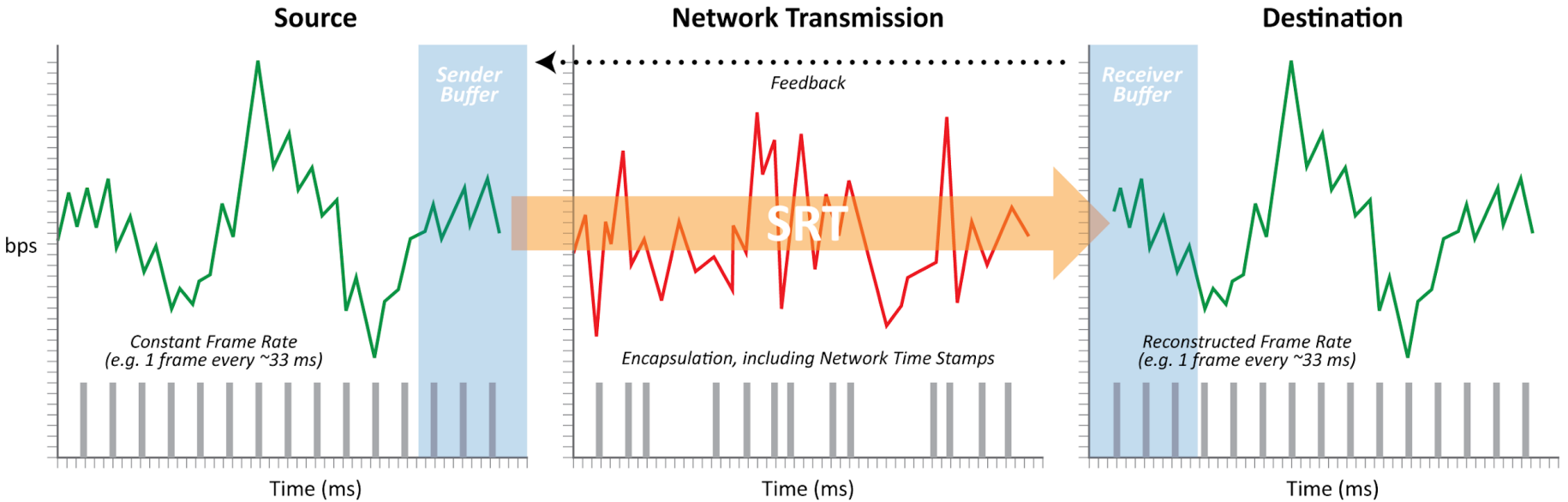
Pristine Quality

Protect against jitter, packet loss and bandwidth fluctuation, so your viewers get the best viewing experience.

Low Latency

Configurable control to deliver low latency video while overcoming network challenges.

Live Stream Integrity & Timing



Thank you!

For more info:

- SRT RFC Draft Proposal
  https://datatracker.ietf.org/doc/draft-sharabayko-mops-srt/
- SRT Technical Overview
  https://github.com/Haivision/srt/files/2489142/SRT_Protocol_TechnicalOverview_DRAFT_2018-10-17.pdf
- SRT Open-source Library
  https://github.com/Haivision/srt
- SRT Alliance
  https://www.srtalliance.org/
- SRT Slack Channel
  https://srtalliance.slack.com/

SRT Packets

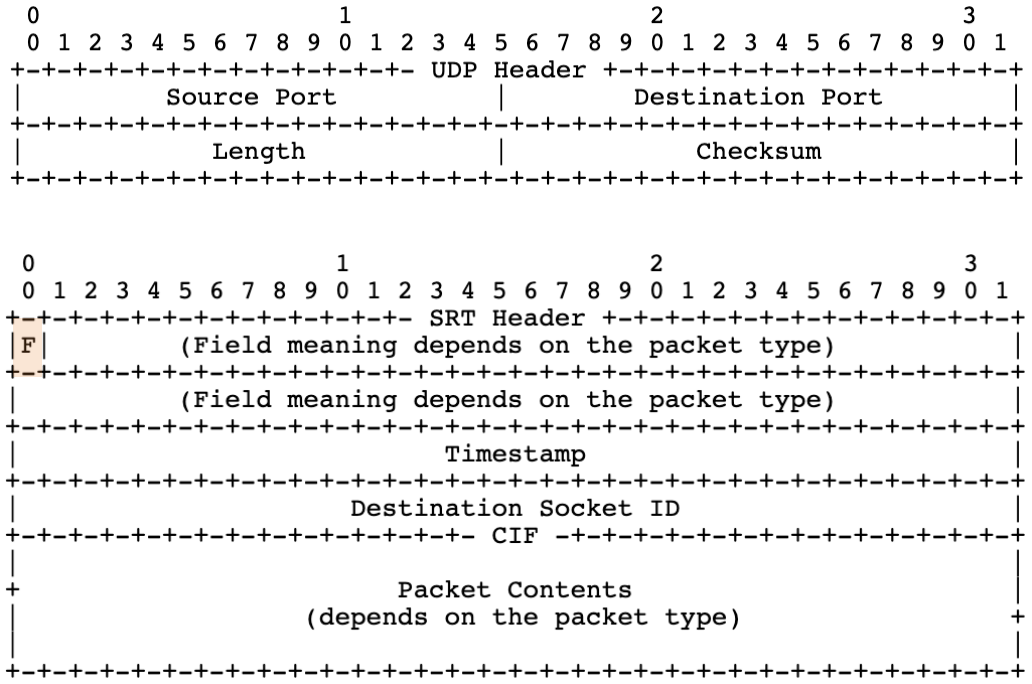
Types of SRT Packets

0: Data Packet

- Data (content to transmit)
- Filtering packet (FEC)

1: Control Packet

- HANDSHAKE
- KEEPALIVE
- ACK
- NAK (Loss Report)
- SHUTDOWN
- ACKACK



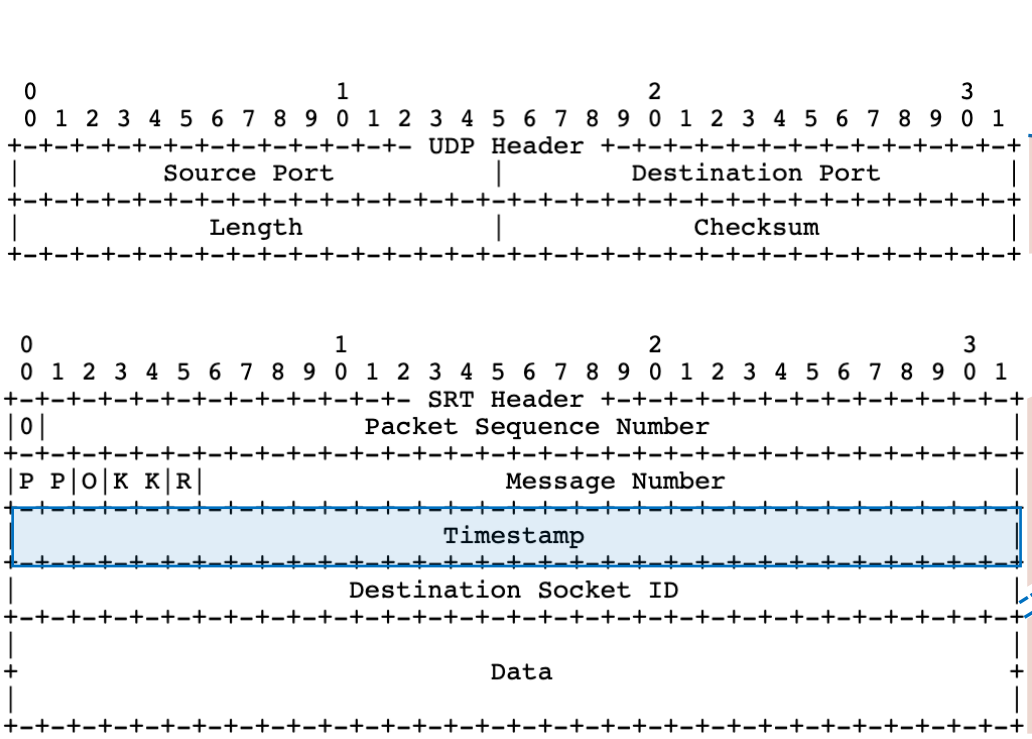
IP Header

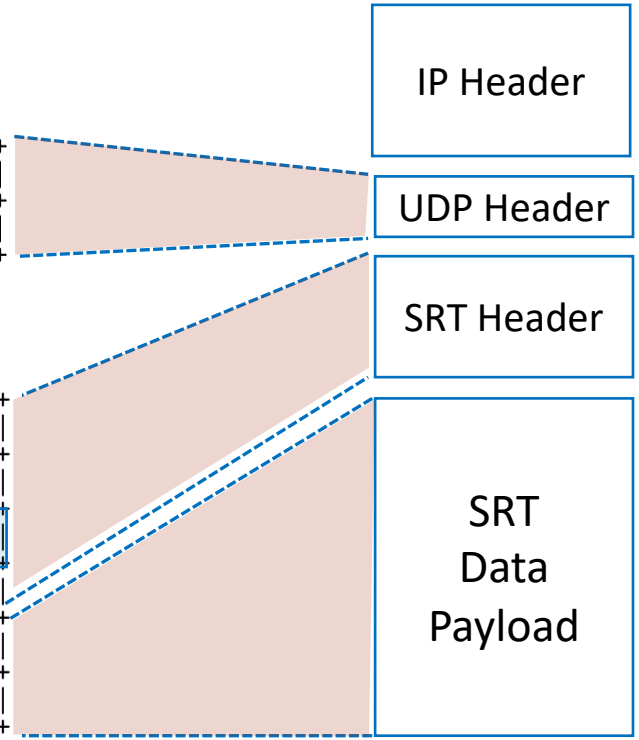
UDP Header

SRT Header

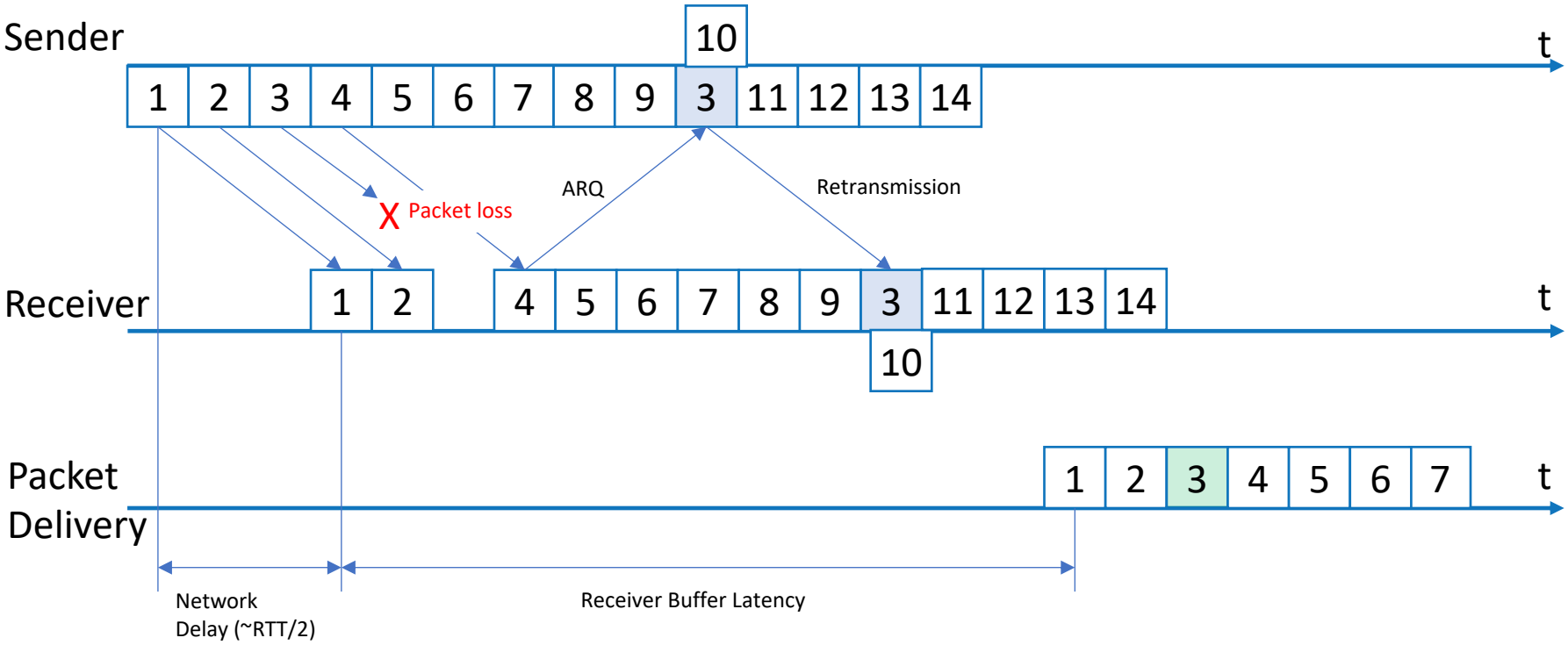
SRT Packet Content

SRT Data Packet

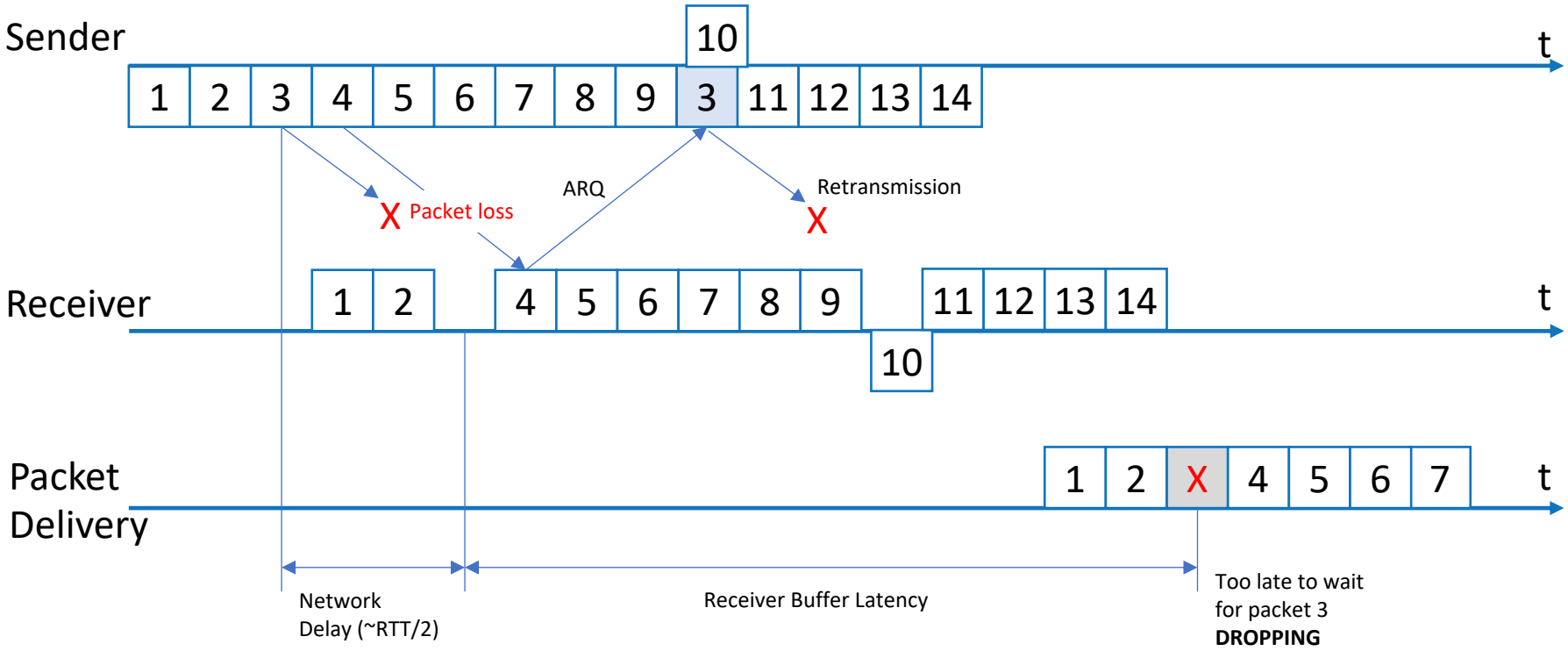




Timestamp Based Packet Delivery (TSBPD)



Too-Late Packet Drop (TL Packet Drop)



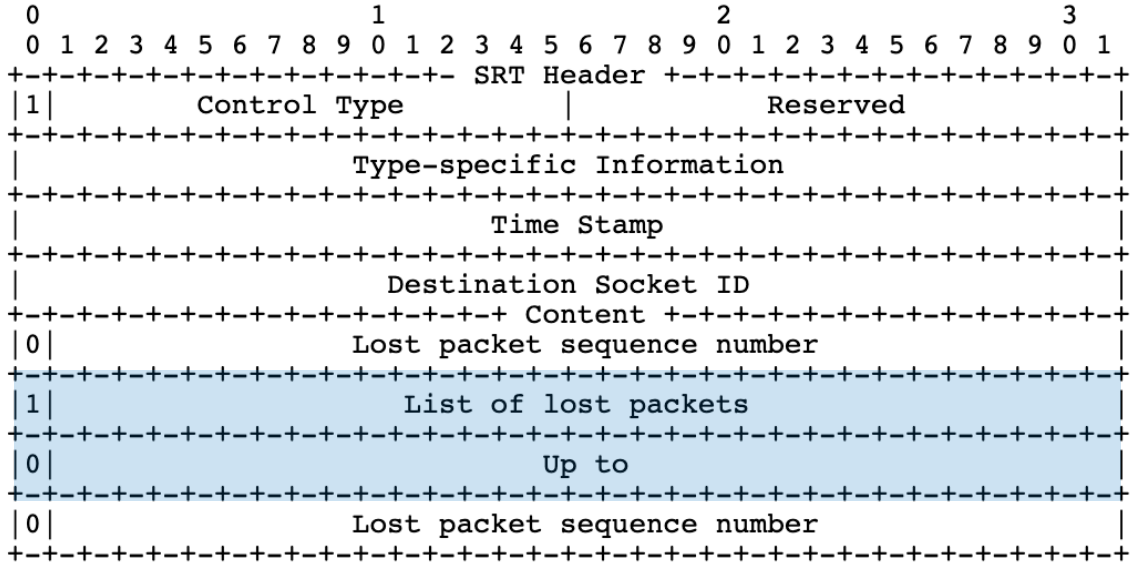
Negative Acknowledgement

Two types of loss reports:

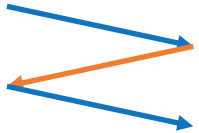
- Loss-triggered NAK reports
- Periodic NAK reports

NAK packet can transmit:

- A single lost packet sequence number;
- A range of sequence numbers of lost packets.



Packet Recovery Options



ARQ

- Loss-triggered NAK reports
- Periodic NAK reports

FEC

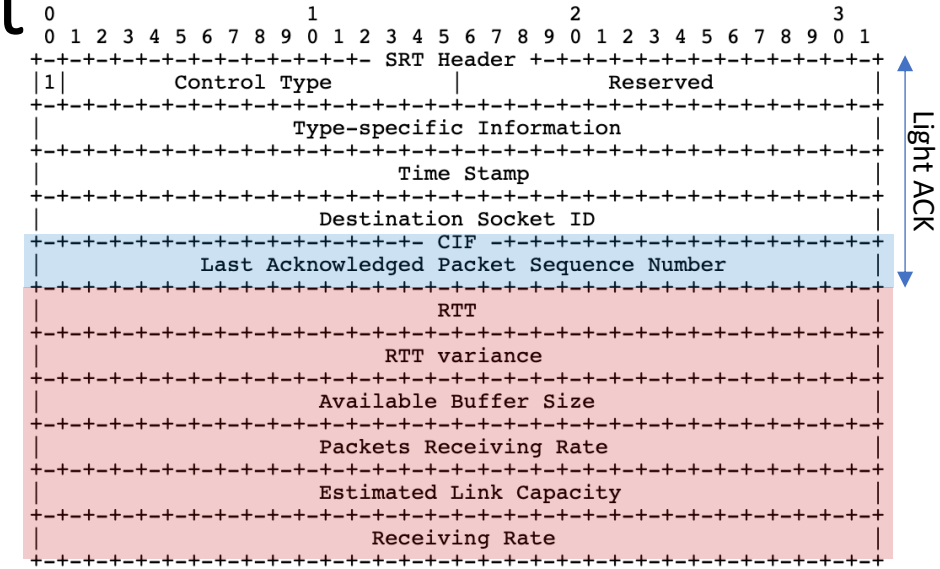
- Error correction packet
- Packet Filter API allows custom implementation

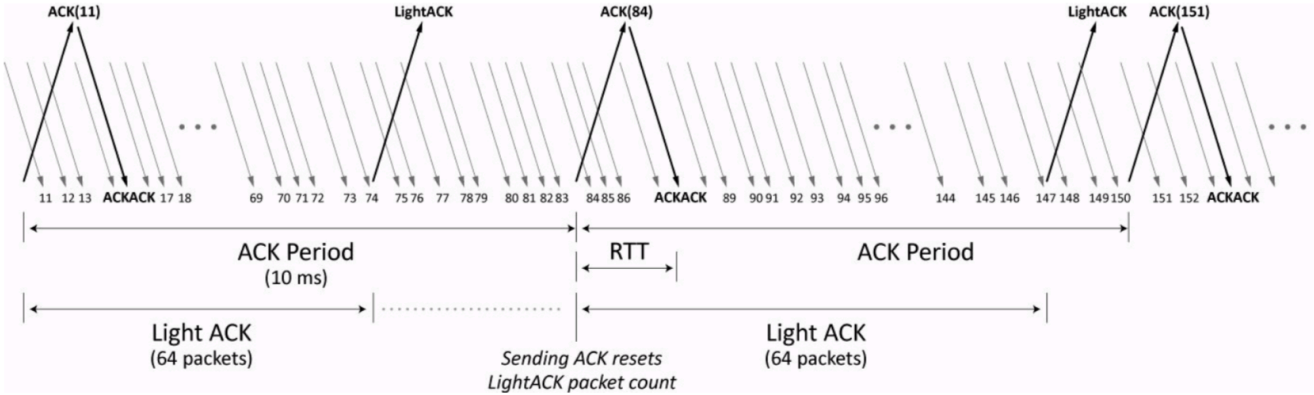
Bonded connections

- Broadcast
- Main-backup
- Load balancing

Positive Acknowledgement

- Acknowledge received packet by its sequence number
- Lost packets block further ACK
- ACK is sent every 10 ms
- Additional info: RTT, Link Capacity, …
- Full ACK is sent every 10ms
- Light ACK is sent on every 64 pkts





Congestion Control

Sender gets from the receiver:

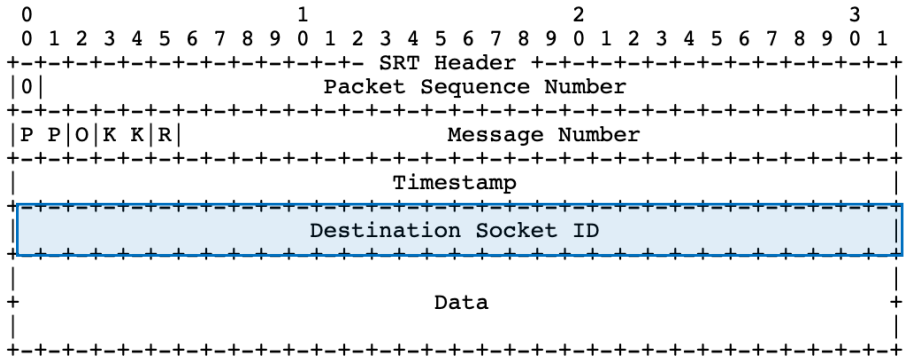
- NAK packet to determine losses
- ACK packets (every 10 ms) to get updates on:
  - RTT
  - RTT variance
  - Estimated link capacity
  - Available receiver buffer
  - Receiving rate
- Congestion control can use this feedback to make decisions!

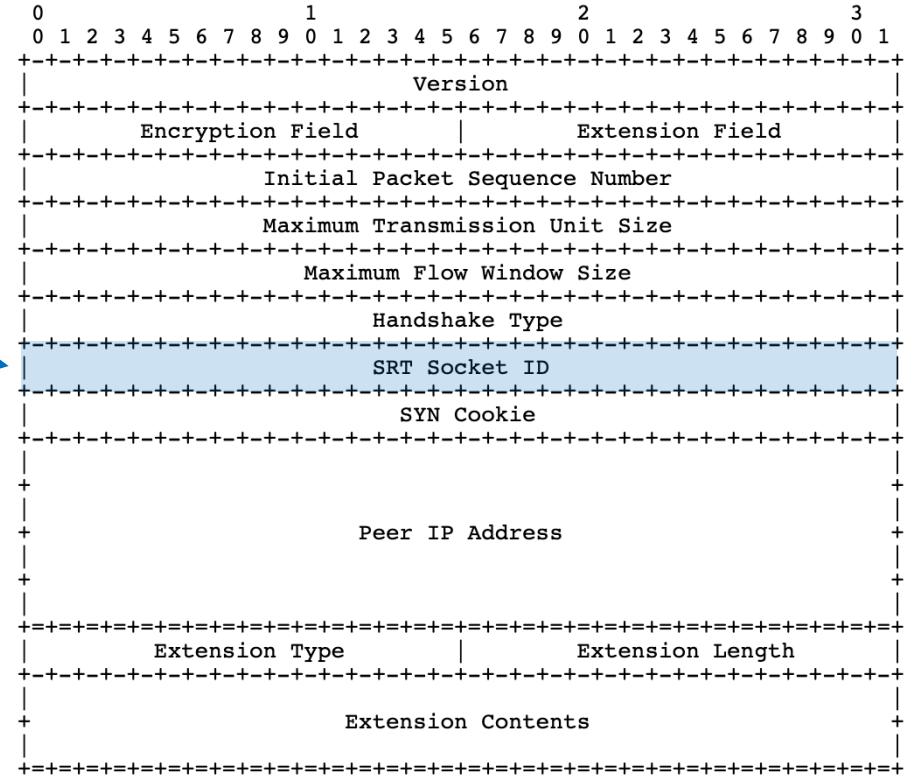
Stream Multiplexing and Bidirectional Transmission

Both peers know each other's IP and **SRT Socket ID** to communicate with data and control packets in both directions.

Handshake Packet

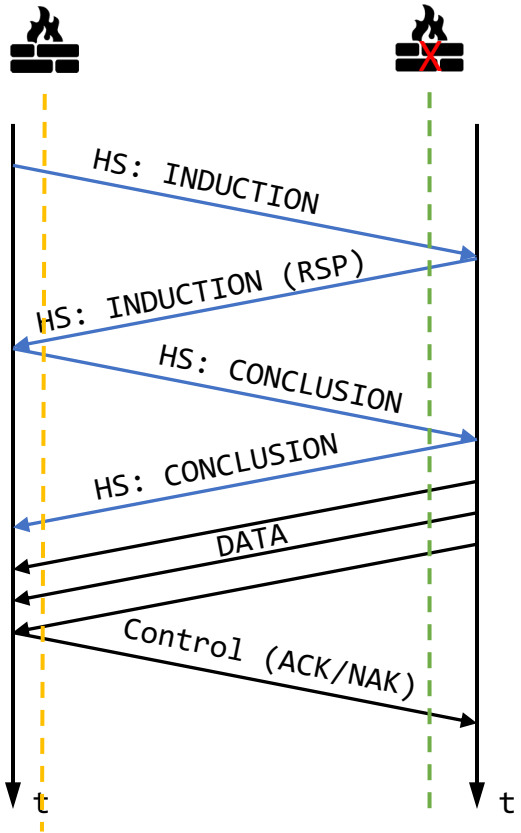
Data Packet



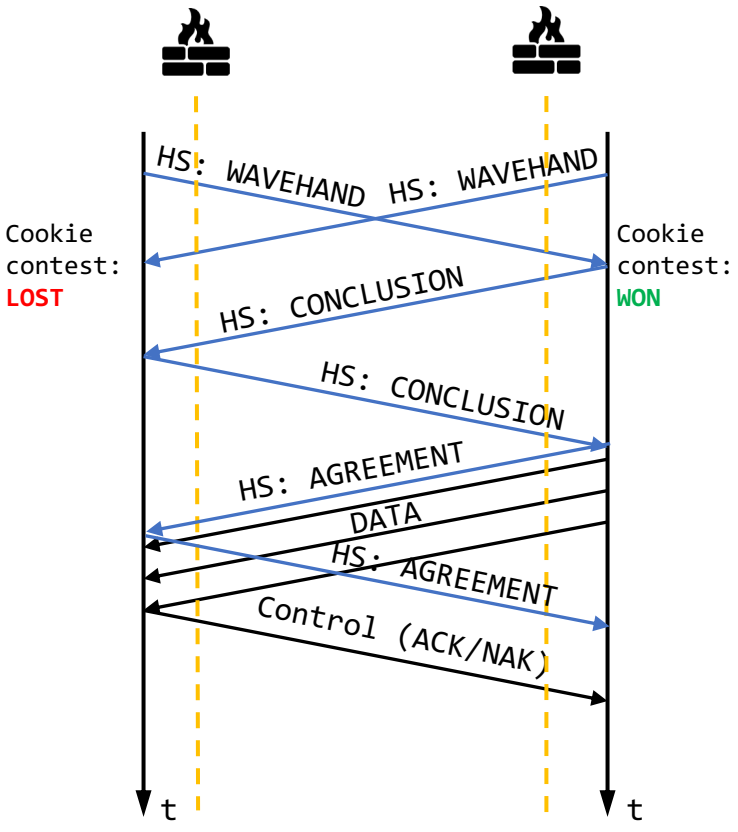


Connection Establishment

Caller-Listener Handshake



Rendezvous Handshake



Firewall Friendly

Establish quality streams from event centers and unknown locations without IT involvement.

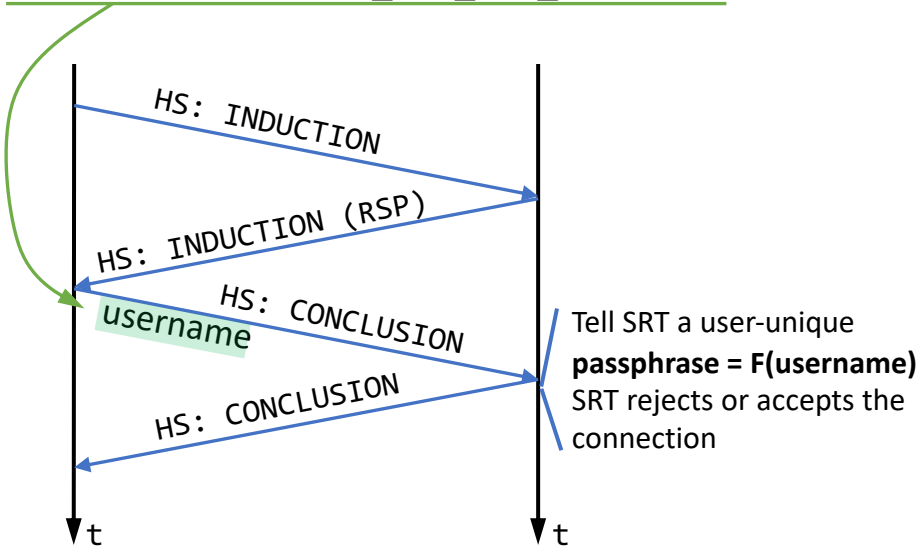
SRT Access Control

The Stream ID free-form value, but there is a recommended convention.

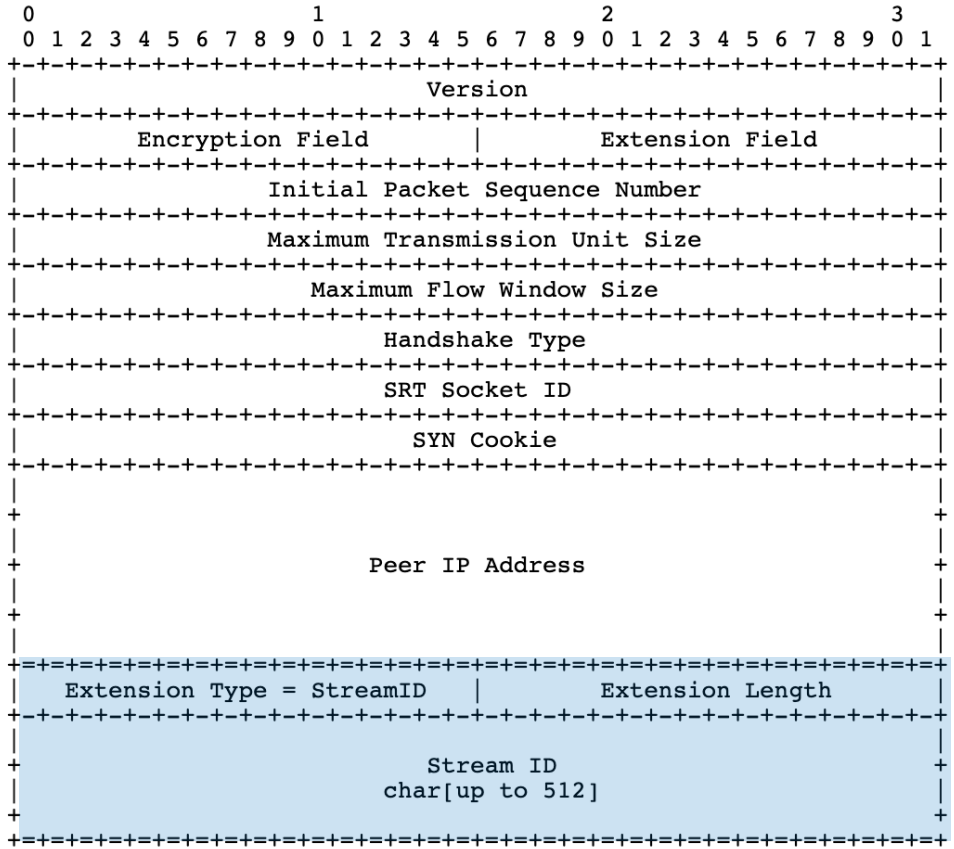
Example content:

requested resource ID

```
#!::u=admin,r=ietf_107_srt_overview
```



Tell SRT a user-unique **passphrase = F(username)**
SRT rejects or accepts the connection



https://github.com/Haivision/srt/blob/master/docs/AccessControl.md

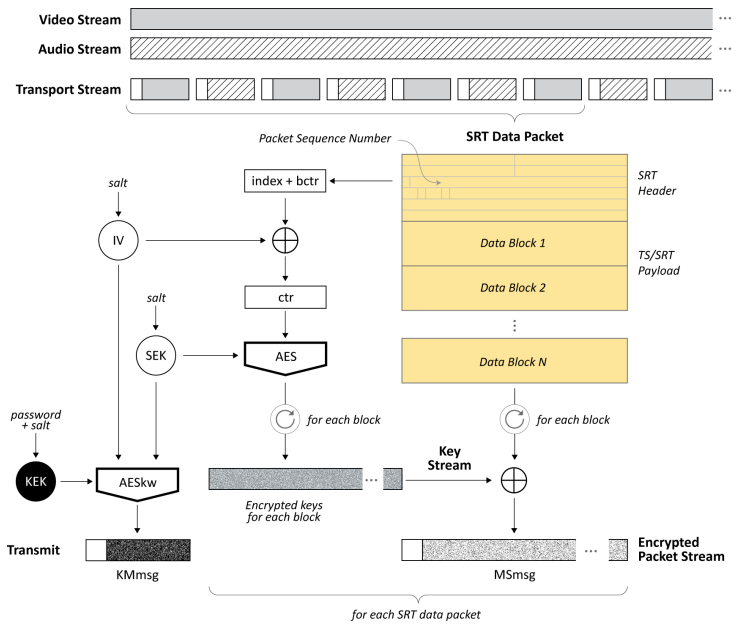
Security & Encryption

- AES 128/192/256-bit encrypted
- Payload encrypted with cipher in AES-CTR mode
- Secret/pass-phrase is not part of the protocol (application layer)

Refer to SRT Technical Overview

https://github.com/Haivision/srt/files/2489142/SRT_Protocol_TechnicalOverview_DRAFT_2018-10-17.pdf

Encryption & Transmission



Reception & Decryption