ECRTP Reordering

draft-koren-avt-ecrtp-reorder-01.txt

Tmima Koren
Patrick Ruddy
Andrew Johnson (Presenting)

AVT WG
IETF 63
Paris, France

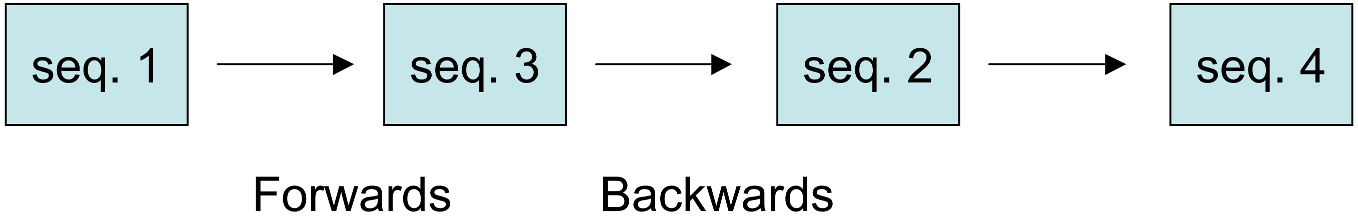
August 2nd, 2005

CRTP and ECRTP

- Reordering not considered
- “twice” algorithm:
  - If you lose a packet then changes are doubled
  - Uses checksums to allow handle packet loss
- ECRTP adds reliability for loss of up to N packets

Packet Loss is re-ordering (kind of)

- Re-ordering is a mix of forward and backward jumps
- A loss can be seen as a jump forward



- Forward jump handling detailed in ECRTP
- This document details how to handle backward jumps

Forward Jumps (twice)

Current Stored Context

Seq.: 1
ID: 101 Delta ID: 1
Time: 5050 Delta Time: 50
…

New packet arrives with a forward skip in sequence

seq.: 3
Checksum: 0x6421

Add deltas twice (3 – 1) and check using the checksum

Backward Jump Problem

Current Stored Context

Seq.: 3
ID: 103 Delta ID: 1
Time: 5150 Delta Time: 50
…

New packet arrives with a backwards skip in sequence

seq.: 2
Checksum: 0x2361

Can’t decompress with this context, discard?

Backward Jump Solution

- Can apply “twice” backwards
- Can’t verify IPv4 ID with checksum
- This is OK if:
  - IPv4 ID is included in update or
  - The delta is known
    - Store RTP seq. of last change

Backward Jump Solution

Store Historical Contexts

Seq.: 1	ID: 101	Delta ID: 1	…
Seq.: 2	Not received		
Seq.: 3	ID: 103	Delta ID: 1	…

New packet arrives with a backwards skip in sequence

seq.: 2
Checksum: 0x2361

Apply “twice” using previous valid context, i.e. seq.: 1

So what can this achieve

- All re-ordering is now a forward jump
- Up to the implementer to limit the number of stored contexts to handle range of re-ordering
- Can jump up to N + 1 forward from any previous context
- Can skip more than N + 1 “forward” as long as checksum works (except …)

IPv4 ID Limitation

- The IPv4 ID is not in the checksum
- This means you can’t skip more than N + 1
- Unless the IPv4 ID is in the received packet

- IPv6 doesn’t have this issue

Further Enhancements

- Jumping more than N + 1 doesn’t necessitate discard
  - Must send a Context Refresh
  - Can store the packet in the context as another out-of-order packet may allow decompression later

Example 1 (within N) Slide A

N = 2

Packet Order	
1	Seq.: 1 ID: 101 Delta ID: 1 TS: 5050 Delta TS: 50 …
	Seq.: 2 Not received
	Seq.: 3 Not received
	Seq.: 4 Not received
	Seq.: 5 Not received
	Seq.: 6 Not received

Example 1 (within N) Slide B

N = 2

Packet Order	
1	Seq.: 1 ID: 101 Delta ID: 1 TS: 5050 Delta TS: 50 …
	Seq.: 2 Not received
	Seq.: 3 Not received
2	Seq.: 4 ID: 104 Delta ID: 1 TS: 5200 Delta TS: 50 …
	Seq.: 5 Not received
	Seq.: 6 Not received

Example 1 (within N) Slide C

N = 2

Packet Order	
1	Seq.: 1 ID: 101 Delta ID: 1 TS: 5050 Delta TS: 50 …
	Seq.: 2 Not received
	Seq.: 3 Not received
2	Seq.: 4 ID: 104 Delta ID: 1 TS: 5200 Delta TS: 50 …
	Seq.: 5 Not received
3	Seq.: 6 ID: 106 Delta ID: 1 TS: 5300 Delta TS: 50 …

Example 1 (within N) Slide D

N = 2

Packet Order	
1	Seq.: 1 ID: 101 Delta ID: 1 TS: 5050 Delta TS: 50 …
	Seq.: 2 Not received
4	Seq.: 3 ID: 103 Delta ID: 1 TS: 5150 Delta TS: 50 …
2	Seq.: 4 ID: 104 Delta ID: 1 TS: 5200 Delta TS: 50 …
	Seq.: 5 Not received
3	Seq.: 6 ID: 106 Delta ID: 1 TS: 5300 Delta TS: 50 …

Example 2 (more than N) Slide A

N = 2

Packet Order	
1	Seq.: 1 ID: 101 Delta ID: 1 TS: 5050 Delta TS: 50 …
	Seq.: 2 Not received
	Seq.: 3 Not received
	Seq.: 4 Not received
	Seq.: 5 Not received
	Seq.: 6 Not received

Example 2 (more than N) Slide B

N = 2

Packet Order	
1	Seq.: 1 ID: 101 Delta ID: 1 TS: 5050 Delta TS: 50 …
	Seq.: 2 Not received
	Seq.: 3 Not received
	Seq.: 4 Not received
2	Seq.: 5 Can’t decompress: send CS, store packet
	Seq.: 6 Not received

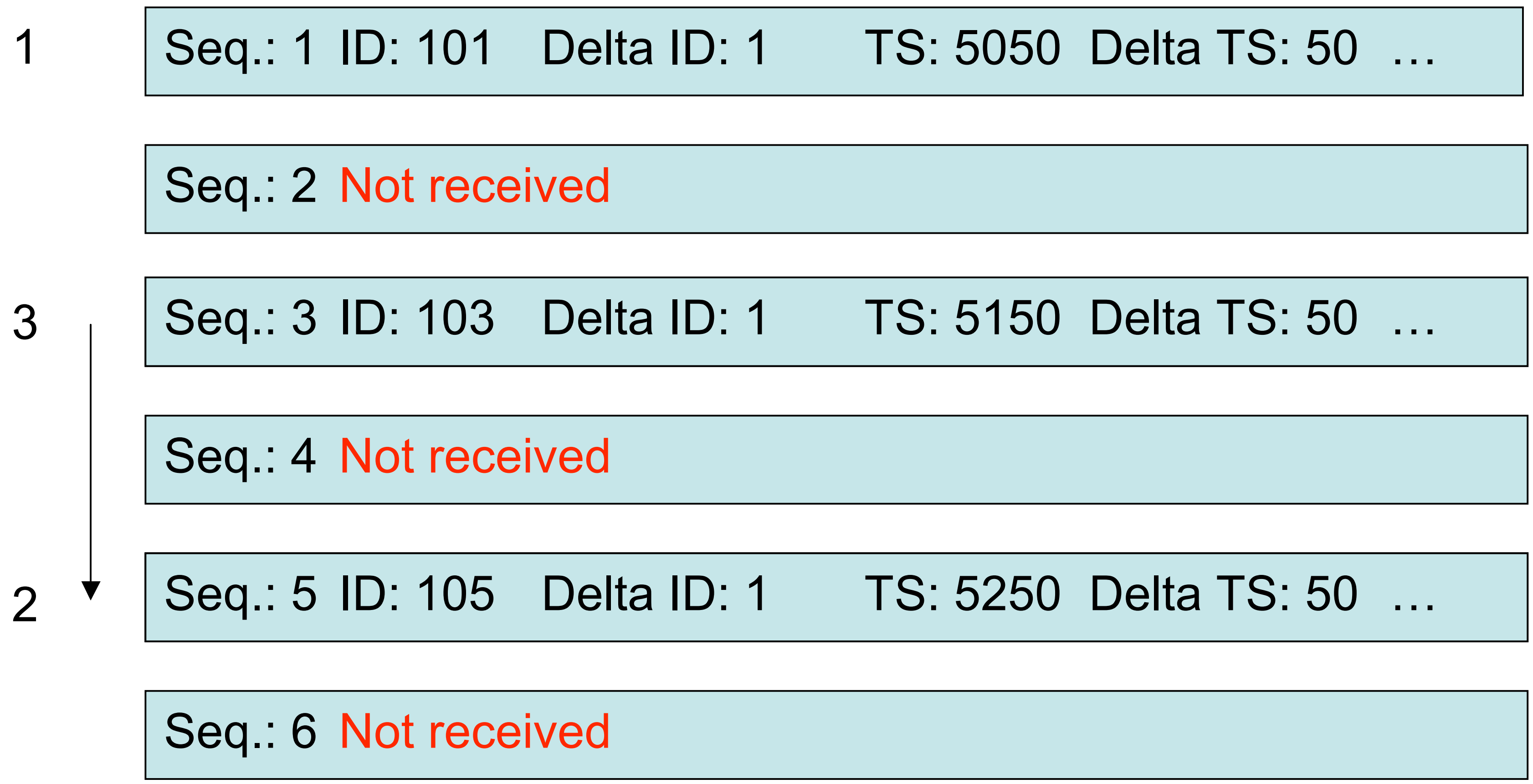
Example 2 (more than N) Slide C

N = 2

Packet Order	
1	Seq.: 1 ID: 101 Delta ID: 1 TS: 5050 Delta TS: 50 …
	Seq.: 2 Not received
3	Seq.: 3 ID: 103 Delta ID: 1 TS: 5150 Delta TS: 50 …
	Seq.: 4 Not received
2	Seq.: 5 Can’t decompress: send CS, store packet
	Seq.: 6 Not received

Example 2 (more than N) Slide D

N = 2



Packet Order	
1	Seq.: 1 ID: 101 Delta ID: 1 TS: 5050 Delta TS: 50 …
	Seq.: 2 Not received
3	Seq.: 3 ID: 103 Delta ID: 1 TS: 5150 Delta TS: 50 …
	Seq.: 4 Not received
2	Seq.: 5 ID: 105 Delta ID: 1 TS: 5250 Delta TS: 50 …
	Seq.: 6 Not received

Example 3 (more than N) Slide A

N = 2, ID in packet

Packet Order

1, ID = 101 Seq.: 1 ID: 101 Delta ID: ? TS: 5050 Delta TS: 50 …

Seq.: 2 Not received

Seq.: 3 Not received

Seq.: 4 Not received

Seq.: 5 Not received

Seq.: 6 Not received

Example 3 (more than N) Slide B

N = 2, ID in packet

Packet Order	
1, ID = 101	Seq.: 1 ID: 101 Delta ID: ? TS: 5050 Delta TS: 50 …
	Seq.: 2 Not received
	Seq.: 3 Not received
	Seq.: 4 Not received
2, ID = 105	Seq.: 5 ID: 105 Delta ID: ? TS: 5250 Delta TS: 50 …
	Seq.: 6 Not received

Example 3 (more than N) Slide C

N = 2, ID in packet

Packet Order	
1, ID = 101	Seq.: 1 ID: 101 Delta ID: ? TS: 5050 Delta TS: 50 …
	Seq.: 2 Not received
3, ID = 103	Seq.: 3 ID: 103 Delta ID: ? TS: 5150 Delta TS: 50 …
	Seq.: 4 Not received
2, ID = 105	Seq.: 5 ID: 105 Delta ID: ? TS: 5250 Delta TS: 50 …
	Seq.: 6 Not received

Questions?

Authors

- Tmima Koren - tmima@cisco.com
- Patrick Ruddy – pruddy@cisco.com
- Andrew Johnson – andrjohn@cisco.com

Please consider making a WG Item