Network Working Group
Request for Comments: 4224
Category: Informational
G. Pelletier
L-E. Jonsson
K. Sandlund
Ericsson
January 2006

RObust Header Compression (ROHC): ROHC over Channels That Can Reorder Packets

Status of This Memo

Copyright Notice

Abstract

RObust Header Compression (ROHC), RFC 3095, defines a framework for header compression, along with a number of compression protocols (profiles). One operating assumption for the profiles defined in RFC 3095 is that the channel between compressor and decompressor is required to maintain packet ordering. This document discusses aspects of using ROHC over channels that can reorder packets. It provides guidelines on how to implement existing profiles over such channels, as well as suggestions for the design of new profiles.

Table of Contents

1. Introduction

RObust Header Compression (ROHC), RFC 3095 [1], defines a framework for header compression, along with a number of compression protocols (profiles). One operating assumption for the profiles defined in RFC 3095 is that the channel between compressor and decompressor is required to maintain packet ordering for each compressed flow. The motivation behind this assumption was that the primary candidate channels considered did guarantee in-order delivery of header-compressed packets. This assumption made it possible to meet the design objectives that were on top of the requirements list at the time when ROHC was being designed, namely to improve the compression efficiency and the tolerance to packet losses.

Since the publication of RFC 3095 in 2001, the question about ROHC operation over channels that do not guarantee in-order delivery has surfaced several times; arguments that ROHC cannot perform adequately over such channels have been heard. Specifically, this has been raised as a weakness when compared to other header compression alternatives, as RFC 3095 explicitly states its inability to operate if in-order delivery is not guaranteed. For those familiar with the details of ROHC and of other header compression schemes, it is clear that this is a misconception, but it can also be easily understood that the wording used in RFC 3095 can lead to such interpretation.

This document discusses the various aspects of implementing ROHC over channels that can reorder header-compressed packets. It explains different ways of implementing the profiles found in RFC 3095, as well as other profiles based on those profiles, over reordering channels. This can be achieved either by ensuring that compressor implementations use compressed headers that are sufficiently robust to the expected possible reordering and/or by modifying decompressor implementations to tolerate reordered packets. Ideas regarding how existing profiles could be updated and how new profiles can be defined to cope efficiently with reordering are also discussed.

In some scenarios, there might be external means (such as a sequence number) to detect and potentially correct reordering. That is, for example, the case when running compression over an IPsec Encapsulating Security Payload (ESP) tunnel. With such external means to detect reordering, the decompressor can be modified to make use of the external information provided, and reordering can then be handled. How to make use of external means to address reordering is, however, out of scope for this document.

2. Terminology

This document uses terminology consistent with RFC 3759 [2], and is in itself only informative. Although it does discuss technical aspects of implementing the ROHC specifications in particular environments, it does not specify any new technology.

ROHC

The term "ROHC" herein refers to the following profiles:

- 0x0001, 0x0002, and 0x0003 defined in RFC 3095 [1];
- 0x0004 for compression of IP-only headers [3];
- 0x0007 and 0x0008 for compression of UDP-Lite headers [4].

The term "ROHC" excludes the following profiles, which are either not affected by reordering or have the assumption of in-order delivery as a fundamental requirement for their proper operation:

- 0x0000 (uncompressed) [1];
- 0x0005 (Link-Layer Assisted (LLA)) [5] and 0x0105 (R-mode extension to LLA) [6];

Reordering

A type of transmission taking place between compressor and decompressor where in-order delivery of header-compressed packets is not guaranteed.

Reordering channel

A connection over which reordering, as defined above, can occur.

Sequentially early packet

A packet that reaches the decompressor before one or several packets of the same context identifier (CID) that were delayed on the link. At the time of the arrival of a sequentially early packet, the packet(s) delayed on the link cannot be differentiated from lost packet(s).

Sequentially late packet

A packet is late within its sequence if it reaches the decompressor after one or several other packets belonging to the same CID have been received, although the sequentially late packet was sent from the compressor before the other packet(s).

Updating packet

A packet that updates the context of the decompressor, e.g., all packets except R-0 and R-1* in RFC 3095 [1].

Non-updating packet

A packet that does not update the context of the decompressor, e.g., only R-0 and R-1* in RFC 3095 [1].

Change packet

A packet that updates one or more fields of the context other than the fields pertaining to the functions established with respect to the sequence number (SN). Specifically, it is a packet that updates fields other than the SN, the IPv4 identifier (IP-ID), the sequence number of an extension header or the RTP timestamp (TS).

3. Applicability of This Document to ROHC Profiles

This document addresses general reordering issues for ROHC profiles. The foremost objectives are to ensure that ROHC implementations do not forward packets with incorrectly decompressed headers to upper layers, as well as to limit the possible increase in the rate of decompression failures or in events leading to context damage, when compression is applied over reordering channels.

3.1. Profiles within Scope

The following sections outline solutions that are generally applicable to profiles 0x0001 (RTP), 0x0002 (UDP), and 0x0003 (ESP) defined in RFC 3095 [1]. Profile 0x0000 (uncompressed) is not affected by reordering, as the headers are sent uncompressed. The solutions also apply to profiles for IP-only (0x0004) [3] and for UDP-Lite (0x0007 and 0x0008) [4]. These profiles are based on the profiles of RFC 3095 [1] and inherently make the same in-order delivery assumption.

3.2. Profiles with Special Considerations

Special considerations are needed to make some of the implementation solutions of sections 6.1 and 6.2 applicable to profiles 0x0002 (UDP) [1], 0x0004 (IP-only) [3], and 0x0008 (UDP-Lite) [4]. For these profiles, the SN is generated at the compressor, as it is not present in headers being compressed. For the least significant bit (LSB) encoding method, the interpretation interval offset (p) is always p = -1 (see section 5.1.1) when interpreting the SN. The SN is thus required to increase for each packet received at the decompressor, which means that reordered packets cannot be decompressed.

3.3. Profiles Incompatible with Reordering

The ROHC LLA profiles defined in RFC 3242 [5] and RFC 3408 [6] have been explicitly designed with in-order delivery as a fundamental requirement to their proper operation. Profiles 0x0005 and 0x0105 can therefore not be implemented over channels where reordering can occur; this document therefore does not apply to these profiles.

4. Background

ROHC was designed with the assumption that packets are delivered in order from compressor to decompressor. This was considered as a reasonable working assumption for links where it was expected that ROHC would be used. However, many have expressed that it would be desirable to use ROHC also over connections where in-order delivery is not guaranteed [7].

4.1. Reordering Channels

The reordering channels that are potential candidates to use ROHC are single-hop channels and multi-hop virtual channels.

A single-hop channel is a point-to-point link that constitutes a single IP hop. Note that one IP hop could be one or multiple physical links. For example, a single-hop reordering channel could be a wireless link that applies error detection and performs retransmissions to guarantee error-free delivery of all data. Another example could be a wireless connection that performs bicasting of data during a handoff procedure.

A multi-hop virtual channel is a virtual point-to-point link that traverses multiple IP hops. A multi-hop virtual channel would typically be an IP tunnel, where compression is applied over the tunnel by the endpoints of the tunnel (not to be confused with single link compression of tunneled packets).

4.2. Robustness Principles of ROHC

Robustness is based on the optimistic approach in the unidirectional and optimistic modes of operation (U/O-mode), and on the secure reference principle in the bidirectional reliable mode (R-mode). Both approaches have different characteristics in the presence of reordering between compressor and decompressor. However, in any mode, decompression of sequentially early packets will generally be handled quite well since they will be perceived and treated by the decompressor as if there had been one or more packet losses.

4.2.1. Optimistic Approach (U/O-mode)

A ROHC compressor uses the optimistic approach to reduce header overhead when performing context updates in U/O-mode. The compressor normally repeats the same update until it is fairly confident that the decompressor has successfully received the information. The number of consecutive packets needed to obtain this confidence is open to implementations, and this number is normally related to the packet loss characteristics of the link where header compression is used (see also [1], section 5.3.1.1.1).

All packet types used in U/O-mode are context updating.

4.2.2. Secure Reference Principle (R-mode)

A ROHC compressor uses the secure reference principle in R-mode to ensure that context synchronization between ROHC peers cannot be lost due to packet losses. The compressor obtains its confidence that the decompressor has successfully updated the context from a packet carrying a 7- or 8-bit Cyclic Redundancy Check (CRC) based on acknowledgements received from the decompressor (see also [1], section 5.5.1.2).

The secure reference principle makes it possible for a compressor to use packets that do not update the context (i.e., R-0 and R-1* [1]).

5. Problem Description

5.1. ROHC and Reordering Channels

This section reviews different aspects of ROHC susceptible of being impacted by reordering of compressed packets between ROHC peers.

5.1.1. LSB Interpretation Interval and Reordering

The least significant bit (LSB) encoding method defined in RFC 3095 ([1], section 5.7) specifies the interpretation interval offset, called p, as follows:

For profiles 0x0001, 0x0003, and 0x0007:

```
p = 1, when bits(SN) <= 4;
p = 2^(bits(SN)-5) - 1 otherwise.
```

The resulting table describing the interpretation interval is as follows:

bits (SN) k	Offset p (reordering)	(2^k-1) - p (losses)
4	1	14
5	0	31
6	1	62
7	3	124
8	7	248
9	15	496

As shown in the table above, the ability for ROHC to handle sequentially late packets depends on the number of bits sent in each packet. For example, a sequentially late packet of type 0 (with either 4 or 6 bits of SN) sets the limit to one packet out of sequence for successful decompression to be possible.

For profiles 0x0002, 0x0004, and 0x0008:

p = - 1, independently of bits(SN).

A value of p = -1 means that the interpretation interval offset can only take positive values and that no sequentially late packet can be decompressed if reordering occurs over the link.

The trade-off between reordering and robustness

The ability of ROHC to handle sequentially late packets is limited by the interpretation interval offset of the sliding window used for LSB encoding. This offset has a very small value for packets with a small number of sequence number (SN) bits, but grows with the number of SN bits transmitted.

For channels where both packet losses and reordering can occur, modifications to the interpretation interval face a trade-off between the amount of reordering and the number of consecutive packet losses that can be handled by the decompressor. If the negative offset (i.e., p) is increased to handle a larger amount of reordering, the value of the positive offset of the interpretation interval must be decreased. This may impact the compression efficiency when the channel has a high loss rate.

This is shown in the figure:

```
  <--- interpretation interval (size is 2^k) ---->
  |-----------------+-------------------------|
Lower              v_ref                       Upper
Bound                                          Bound
  <--- reordering --> <--------- losses --------->
   max delta(SN) = p   max delta(SN) = (2^k-1) - p
```

where v_ref is the reference value as per [1], section 4.5.1.

In practice, the maximum variation in SN value (max delta(SN)) due to reordering that can be handled will normally correspond to the maximum number of packets that can be reordered. The same applies to the maximum number of consecutive packet losses covered by the robustness interval.

Timer-based compression of RTP TS (see [1], section 4.5.4) provides means to reduce the number of timestamp bits needed in compressed headers after longer gaps in the packet stream (e.g., for an audio stream, this is typically due to silence suppression). To use timer-based compression, an upper limit on the inter-arrival jitter must be reliably estimated by the compressor. It should be noted that although the risk of reordering of course means there is a more significant jitter on the path between the compressor and the decompressor, there are no special reordering considerations for timer-based compression. It all still boils down to the task of estimating the jitter, requiring channel characteristics knowledge at the compressor, and/or jitter estimation figures received from the decompressor.

5.1.2. Reordering of Packets in R-mode

5.1.2.1. Updating Packets

The compressor always adds references in the sliding window for all updating packets sent. The compressor removes values older than values for which it has received an acknowledgement to shrink the window and thereby increase the compression efficiency.

The decompressor always updates the context when receiving an updating packet and uses the new reference for decompression. Acknowledgements are sent to allow the compressor to shrink its sliding window.

Reordering between updating packets

The decompressor can update its context from the reception of a sequentially late updating packet. The decompressor reference is then updated with a value that is no longer in the sliding window of the compressor. This "missing reference" can be caused by reordering when operating in R-mode.

The result is that the compressor and the decompressor lose synchronization with each other. When the decompressor acknowledges the sequentially late packet, the compressor might already have discarded the reference to this sequence number, and continue to compress packets based on more recent references (in packet arrival time). Decompression will then be attempted using the wrong reference.

5.1.2.2. Non-Updating Packets

Reordering between non-updating packets only

A non-updating packet that reaches the decompressor out of sequence only with respect to other non-updating packets can always be decompressed properly.

Reordering between non-updating packets and updating packets

When a non-updating packet is reordered and becomes sequentially late with respect to an updating packet, the decompressor may have already updated the context with a new reference when the late packet is received. It is thus possible for a non-updating packet to be decompressed based on the wrong reference because of reordering when operating in R-mode.

Since decompression of non-updating packets cannot be verified, this can lead to a packet erroneously decompressed to be forwarded to upper layers.

5.1.3. Reordering of Packets in U/O-mode

Reordering between non-change packets only

When only non-change packets are reordered with respect to each other, decompression of sequentially late packets is limited by the offset p of the interpretation interval (see section 5.1.1). Decompression of a sequentially late packet with SN = x is possible if the value of the SN of the packet that last updated the context was less than or equal to x + p.

Problems occur if context(SN) has increased by more than p with respect to field(SN) carried within the packet to decompress.

This means that for a well-behaved stream with a constant unit increase in the RTP SN, a packet can arrive up to p packets out of sequence and still be correctly decompressed. Otherwise, it cannot be properly decompressed. It also means that if the compressor sends two consecutive packets with SN(packet1)=100 and SN(packet2)=108 when p=7, packet1 cannot be decompressed if it arrives even one packet late due to reordering.

Reordering involving change packets

When a packet is reordered and becomes sequentially late with respect to a change packet, decompression of the late packet may eventually fail, as the context information required for successful decompression may not be available anymore.

Decompression can always be verified since all U/O-mode packet types are context updating. Consequently, a failure to decompress a packet that is caused by reordering can be detected, and context invalidation due to reordering can thus be avoided. The risk of forwarding incorrectly decompressed packets to upper layers is therefore small when operating in U/O-mode. For channels known to reorder packets, U/O-mode should therefore be the preferred mode of operation. The additional risk of losing context synchronization, or for erroneous packet to be delivered to upper layers, is limited.

5.1.4. Reordering on the Feedback Channel

For R-mode, upon reception of an acknowledgement, the compressor searches the sliding window to locate an updating packet with the corresponding SN; if it is not found, the acknowledgement is invalid and is discarded ([1], section 5.5.1.2). In other words, feedback received out of order either is still useful or is discarded.

In U/O-mode, if the compressor updates its context based on feedback, the same logic as for R-mode applies in practice.

Reordering on the feedback channel has thus no impact in either mode.

5.1.5. List Compression

ROHC list compression is an additional compression scheme for RTP contributing source (CSRC) lists and IP extension header chains. The base is called table-based item compression, and it is almost completely independent from the rest of the ROHC compression logic. Therefore, this part of the scheme does not exhibit any special vulnerabilities when it comes to reordering, assuming a reasonable optimistic approach is used in U/O-mode. Specifically, it does not suffer significantly from the "missing reference" problem when operating in R-mode.

On top of the table-based item compression mechanism, an additional compression technique may be used, called reference based list compression. Reference based list compression however has a logic that is similar to the rest of the ROHC compression logic, and therefore it suffers from similar reordering vulnerabilities, especially the "missing reference" problem of R-mode. Note, however, that the generation identifier used in U/O-mode makes that scheme more robust to reordering.

When using list encoding type 1, 2, or 3, which makes use of reference lists, decompression will succeed only if all individual items are known by the decompressor, along with the correct reference list required to properly decompress the packet. List compression using the "Generic scheme", also known as "Encoding type 0", is not using reference based list compression, and type 0 decompression will thus succeed as long as all individual items are known by the decompressor. Because of this, type 0 list compression should be the preferred method used when operating over reordering channels.

5.1.6. Reordering and Mode Transitions

Transition from U/O-mode to R-mode

This transition can be affected by reordering if a packet type 0 (UO-0) is reordered and delayed by at least one round-trip time (RTT). If the decompressor initiates a mode change request to R-mode in the meantime, the reordered UO-0 packet may be handled as an R-0 packet; it can be erroneously decompressed and forwarded to upper layers. This is because the decompressor can switch to R-mode as soon as it sends the acknowledgement Ack(SN, R) to the compressor (see also [1], section 5.6).

Transition from R-mode to U/O-mode

A similar situation as above can occur during this transition. However, because the outcome of the decompression is always verified using a CRC verification in U/O-mode, the reordered packet will most likely fail decompression and will be discarded.

The above situation, although it is not deemed to occur frequently, is still possible; thus, mode transitions from U/O-mode to R-mode should be avoided when reordering can occur.

5.2. Consequences of Reordering

The context updating properties of the packets exchanged between ROHC peers are the most important factors to consider when deriving the impacts of reordering. For this reason, the robustness properties of the U/O-mode and of the R-mode are affected differently.

The effects of reordering on ROHC can be summarized as follows:

- Functionality incompatible with reordering;
- Increased probability of context damage (loss of synchronization);
- Increased number of decompression failures - Detected (U/O/R-mode);
- Increased number of decompression failures - Undetected (R-mode).

5.2.1. Functionality Incompatible with Reordering

There is one optional ROHC function that cannot work in the presence of reordering between ROHC peers.

The ROHC segmentation scheme (see [1], section 5.2.5) relies entirely on the in-order delivery of each segment, as there is no sequencing information in the segments. A segmented packet for which one (or more) segment is received out of order cannot be decompressed, and it is discarded by the decompressor. Therefore, segmentation should not be used if there can be reordering between the ROHC peers.

The use of this optional feature is open to implementations and is local to the compressor only; it does not impact the decompressor.

5.2.2. Context Damage (Loss of Synchronization)

Reordering of packets between ROHC peers can impact the robustness properties of the optimistic approach (U/O-mode) as well as the reliability of the secure reference principle (R-mode).

The successful decompression of a sequentially late change packet (U/O-mode) and/or updating packet (R-mode) can update the context of the decompressor in a manner unexpected by the compressor. This can lead to a loss of context synchronization between the ROHC peers.

5.2.3. Detected Decompression Failures (U/O/R-mode)

Reordering of packets between ROHC peers can lead to an increase in the number of decompression failures for context updating packets (see sections 5.1.2.1 and 5.1.3). Fortunately, as the outcome of the decompression of updating packets can be verified, the decompressor can reliably detect decompression failures, including those caused by reordering, and discard the packet. Note that local repairs, subject to the limitations stated in [1] section 5.3.2.2.3, can still be performed.

5.2.4. Undetected Decompression Failures (R-mode only)

Reordering of packets between ROHC peers can lead to an increase in the number of decompression errors for non-updating packets. For R-mode, decompression of R-0 and R-1* packets cannot be verified. If reordering occurs and decompression is performed using the wrong secure reference (see section 5.1.2.1 and 5.1.2.2), the decompressor cannot reliably detect such errors. As a result, erroneous packets may be forwarded to upper layers.

6. Making ROHC Tolerant against Reordering

This section describes different approaches that can improve the performance of ROHC when used over reordering channels and minimize the effects of reordering. Examples are provided to guide implementers and designers of new profiles. The solutions target either the properties of ROHC implementations or the specification of profiles. This is covered by sections 6.1 and 6.2, respectively.

6.1. Properties of ROHC Implementations

Existing ROHC profiles can be implemented with the capability to properly handle packet reordering. The methods described in this section conform with, and thus do not require any modifications to, the ROHC specifications within scope of this document (see section 3). Specifically, the methods presented in this section can be implemented without any impairment to interoperability with other ROHC implementations that do not use these methods.

The methods suggested here may, however, lower the compression efficiency, and these modifications should not be used when reordering is known not to occur. Some of these methods aim to increase the decompression success rate at the decompressor, while others aim to avoid context damage that would cause a loss of context synchronization between compressor and decompressor.

The methods proposed are each addressing specific issues listed in section 5 and can be combined to achieve better robustness against reordering.

6.1.1. Compressing Headers with Robustness against Reordering

The methods described in this section are methods local only to the compressor implementation. They can be used without modifications or impact to the decompressor.

6.1.1.1. Reordering and the Optimistic Approach

The optimistic approach is affected by the reordering characteristics of the channel when operating over a reordering channel. Compressor implementations should therefore adjust their optimistic approach strategy to match both packet loss and reordering characteristics.

For example, the number of repetitions for each context update can be increased. The compressor should ensure that each update is repeated until it is reasonably confident that at least one change packet in the sequence of repetitions has reached the decompressor before the first packet sent after this sequence.

6.1.1.2. Reordering and the Secure Reference Principle

Fundamental to the secure reference principle is that only values acknowledged by the decompressor can be used as reference for compression. In addition, some of the packet types used in R-mode do not include a CRC over the original uncompressed header, and the decompressor has no means to verify the outcome of the decompression.

Decompression of non-updating packet types thus entirely relies on the cumulative effect of previous updates to the secure reference, and the compressed data is based on the current value of the reference. This reference must be synchronized between ROHC peers. For R-0 and R-1* packets, the reception of the encoded bits applied to the secure reference is sufficient for correct decompression, but only when in-order delivery between ROHC peers is guaranteed.

Avoiding the "missing reference" problem (section 5.1.2.1)

> A compressor implementation can delay the advance in the sliding window to a reference acknowledged by the decompressor, until it has confidence that no acknowledgement for any of the values that could be discarded can be received. This confidence can be based on the maximum delay that reordering can introduce over the channel.

6.1.1.3. Robust Selection of Compressed Header

Packet formats can be chosen with an interpretation interval for the LSB encoded sequence number that allows for larger negative offsets (see section 5.1.1). This provides the capability to decompress sequentially late packets with a greater amount of reordering.

To achieve this, the compressor should be implemented conservatively in terms of the choice of packet types to send, by transmitting packets with more sequence number bits. As shown in the table in section 5.1.1, using 8 bits of SN allows a packet to be decompressed when the reordering leads to up to 7 units in sequence number variation (i.e., delta(SN)). Increasing the number of SN bits (i.e., using a larger SN_k [1]) transmitted will make ROHC even more tolerant to reordering.

For example, a conservative compressor implementation could use the packet types as shown in the table below:

Optimal Packet Type (without reordering)	Alternative Packet Type (reordering possible)
UO-0	UOR-2*-ext0
R-0	R-1*-ext0
R-0-CRC	UOR-2*-ext0
R-1*	R-1*-ext0
UO-1	UOR-2-ext0
UO-1-TS	UOR-2-TS-ext0
UO-1-ID	UO-1-ID-ext3 (with S=1) UOR-2-ID-ext0
UOR-2*	UOR-2*-ext0

Such a compressor implementation would thus always be sending at least 3 octets (R-mode) or 4 octets (U/O-mode). This is a trade-off when compared to the 1 octet that can be sent by a more aggressive implementation operating on a channel with no reordering.

Note that since the interpretation interval for profiles 0x0002, 0x0004, and 0x0008 is always p = -1 independently of bits(SN), the methods suggested in this section will not work for these profiles unless this value is modified (section 6.2.1).

6.1.2. Implementing a Reordering-Tolerant Decompressor

The methods described in this section are methods local only to the decompressor implementation. They can be used without modifications or impact to the compressor.

6.1.2.1. Decompressor Feedback Considerations

Reducing the feedback rate when the flow behaves linearly

The decompressor should reduce its feedback rate when a large number of UOR-2 packets with extensions are received, when the flow behaves linearly (i.e., when only fields pertaining to the functions established with respect to the sequence number are changing).

In particular, if the compressor implementation makes a more conservative selection of packet types (section 6.1.1.3) in order to handle reordering, the decompressor should try to avoid sending more feedback than it would for the case where the more optimal packet types are used. This can be useful to minimize the usage of the feedback channel, thereby improving efficiency of the link.

Note that even if the decompressor does not make this adjustment to its feedback rate, packet losses or context damages will not increase.

Acknowledgements and sequentially late packets

Reordered feedback (or feedback for packets received out of order) will not cause problems (see section 5.1.4). However, the decompressor should not send acknowledging feedback for a packet that can be identified as being sequentially late (e.g., based on the sequence number of the packet), as the current state of the context will better reflect the compressor context than the content of the reordered packet.

6.1.2.2. Considerations for Local Repair Mechanisms

When decompression fails, and if reordering can be assumed to be the cause of this failure, subsequent decompressions may be attempted for sequentially late packets by going backward in the interpretation interval (as opposed to moving forward for local repair). If one of the decompression attempts is successful, the late packet may be passed on to upper layers with or without updating the decompressor context. If the subsequent decompression attempt fails, the packet should be handled according to [1] section 5.3.2.2.3.

6.2. Specifying ROHC Profiles with Robustness against Reordering

6.2.1. Profiles with Interpretation Interval Offset p = -1

New revisions of profiles 0x0002 (UDP) [1], 0x0004 (IP-only) [3], and 0x0008 (UDP-Lite) [4] should redefine how the value of the offset p is determined, and use the same algorithm as in profile 0x0001 [1] instead of p = -1 independently of bits(SN) (section 5.1.1).

While such a change would make these updated profiles slightly less robust to packet losses, they would still be no less robust than profile 0x0001.

6.2.2. Modifying the Interpretation Interval Offset

The interpretation interval offset p could be modified for existing profiles to handle reordering while improving the compression efficiency when compared to the solution in section 6.1.1.3.

6.2.2.1. Example Profile for Handling Reordering

The value of the interpretation interval offset p can be adjusted to achieve a robustness against reordering similar to the effect of selecting packet types as suggested in section 6.1.1.3.

Consider a scenario where robustness against packet losses is kept a priority, and for which of a value p=7 is deemed enough. In this case, a ratio where the positive offset is about twice as large as the negative offset can be used. This leaves a value of p = 2^k/ 3.

The resulting values are shown in the following table:

bits (SN) k	Offset p (reordering)	Positive range (losses)
4	5	10
5	10	21
6	21	42
7	42	85
8	85	170
9	170	341

Using this value for p, a fair amount of reordering can be handled without having to send UOR-2 packets most of the time. The trade-off is that this is at the expense of robustness against packet losses.

6.2.2.2. Defining the Values of p for New Profiles

As described in RFC 3095 [1], the interpretation interval when sending k bits of SN is defined as follows:

```
f(v_ref, k) = [v_ref - p, v_ref + (2^k - 1) - p]
```

The negative bound (v_ref - p) limits the ability to handle reordering, and the positive bound (v_ref + (2^k - 1) - p) limits the ability to handle packet losses.

Adjusting p will increase one of these ranges, while the other range will decrease. This trade-off between the capability to handle reordering and packet losses, including how these correlate with each other, should be considered in a ROHC profile that is meant to handle reordering.

For example, if it is desirable for a profile to be as robust against reordering (negative range) and against packet losses (positive range), this range can be made equal by setting p near (2^k / 2).

7. Security Considerations

This document does not include additional security risks to [1]. In addition, it may lower risks related to context damage in R-mode with injected packets when sequentially late packets do not update the context (section 6.1.2.1).

8. Acknowledgements

Thanks to the committed WG document reviewers, Carl Knutsson and Mark West, for their review efforts. Thanks also to Aniruddha Kulkarni, Ramin Rezaiifar, and Gorry Fairhurst for their constructive comments.

9. Informative References

[1] Bormann, C., Burmeister, C., Degermark, M., Fukushima, H., Hannu, H., Jonsson, L-E., Hakenberg, R., Koren, T., Le, K., Liu, Z., Martensson, A., Miyazaki, A., Svanbro, K., Wiebke, T., Yoshimura, T., and H. Zheng, "RObust Header Compression (ROHC): Framework and four profiles: RTP, UDP, ESP, and uncompressed", RFC 3095, July 2001.

[2] Jonsson, L-E., "RObust Header Compression (ROHC): Terminology and Channel Mapping Examples", RFC 3759, April 2004.

[3] Jonsson, L-E. and G. Pelletier, "RObust Header Compression (ROHC): A Compression Profile for IP", RFC 3843, June 2004.

[4] Pelletier, G., "RObust Header Compression (ROHC): Profiles for User Datagram Protocol (UDP) Lite", RFC 4019, April 2005.

[5] Jonsson, L-E. and G. Pelletier, "RObust Header Compression (ROHC): A Link-Layer Assisted Profile for IP/UDP/RTP", RFC 3242, April 2002.

[6] Liu, Z. and K. Le, "Zero-byte Support for Bidirectional Reliable Mode (R-mode) in Extended Link-Layer Assisted RObust Header Compression (ROHC) Profile", RFC 3408, December 2002.

[7] Ash, J., Goode, B., Hand, J., and R. Zhang, "Requirements for Header Compression over MPLS", RFC 4247, November 2005.

Authors' Addresses

Ghyslain Pelletier
Ericsson AB
Box 920
SE-971 28 Lulea, Sweden

Phone: +46 8 404 29 43
Fax: +46 920 996 21
EMail: ghyslain.pelletier@ericsson.com

Lars-Erik Jonsson
Ericsson AB
Box 920
SE-971 28 Lulea, Sweden

Phone: +46 8 404 29 61
Fax: +46 920 996 21
EMail: lars-erik.jonsson@ericsson.com

Kristofer Sandlund
Ericsson AB
Box 920
SE-971 28 Lulea, Sweden

Phone: +46 8 404 41 58
Fax: +46 920 996 21
EMail: kristofer.sandlund@ericsson.com

Acknowledgement

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).