INTERNET-DRAFT
Intended Status: Standard
Expires: September 2019

T. Herbert
Quantonium

**March 11, 2019**

# Simple Segment Routing Header
draft-herbert-simple-sr-00

## Abstract

This specification defines a simple extension header format for Segment Routing based on the current definition of the Segment Routing extension header defined for IPv6. A Segment Identifier type field is added so that the segment list might contain values other than IPv6 addresses. Optional TLVs in the segment routing header are eliminated; Destination options that precede the routing header are sufficient. Two new destination options are defined: one for Routing header security and another to specify that certain destinations should process certain options.

## Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/1id-abstracts.html

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

Table of Contents

## 1 Introduction

This specification defines a simplified segment routing header and generalizes some aspects of segment routing to be applicable to other types of routing headers. The segment routing header is defined in [SRHV6].

The following modifications and additions are defined:

* A Segment Identifier type field in the Segment routing header.

  This field indicates the type of elements in the segment routing list and also indicates the length of each element. Segment Identifier types are defined for IPv6 and IPv4 addresses, as well as types for indices into tables that would map a Segment Identifier to an address. The concept of types for Segment Identifier is a generalization of Segment Routing header compression defined in [SRCOMP].

* Eliminate options (TLVs) from Segment Routing header.

  Options pertaining to Segment Routing, or more generally any type of Routing header, may be set in Destination Options that precede the Routing header.

* Routing header security Destination option.

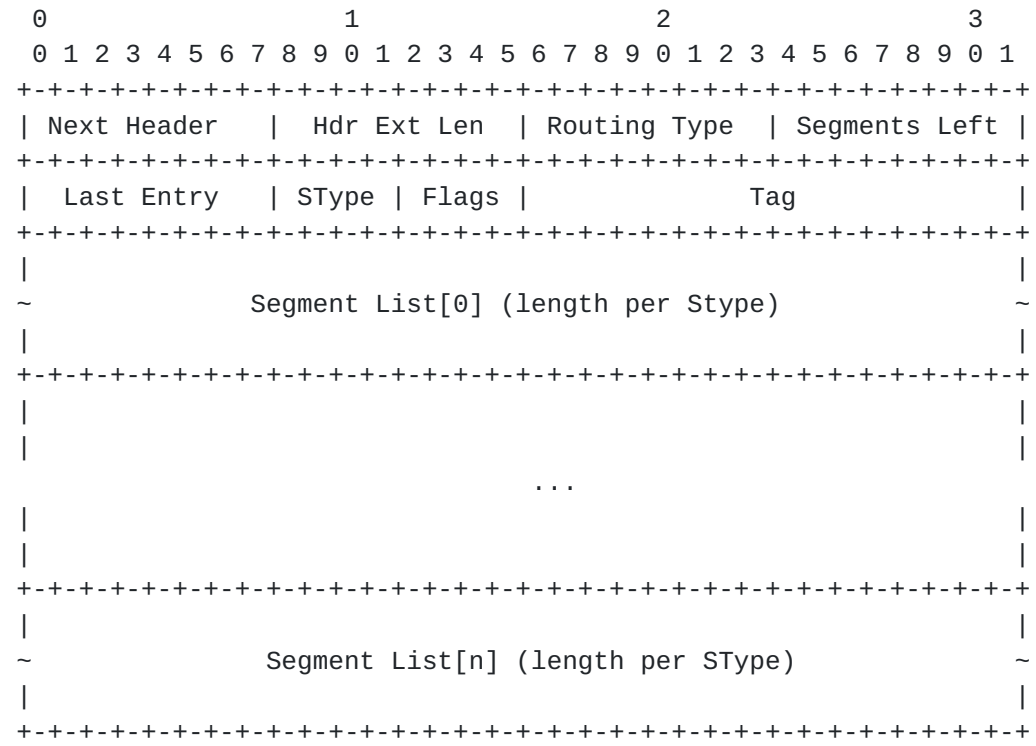
  This option provides an extensible method for verifying security of a routing header. An HMAC option is defined that provides the same functionality as the HMAC TLV defined in [SRV6EH] (except that it is generic to work with all types of Routing headers).

* Process per segment Destination option.

  The purpose of this option is to allow a node to indicate that certain Destination options are to be processed only by certain nodes in the Segment List. This is a generalization of the option described in [SRENDOPT].

## 2 Simple segment routing header format

The format of the simple segment routing header is shown below.




The format is based on that described in [SRV6EH]. Modified or added fields are:

- SType: Type of segment identifiers. Possible values are listed below.

- Segment List[]. Each element of the segment list contains a segment identifier with the type indicated by SType. The length of each identifier is implied by the SType.

Note that the optional TLVs section is not present in the simplified format. The rest of the fields in the format retain the same meaning an format as described in [SRV6EH].

### 2.1 SType values

The following are the SType values:

- 0: IPv6 address, 128 bit value

- 1: IPv6 identifier, 64 bit value

- o 2: IPv6 locator, 64 bit value
- o 3: IPv4 address, 32 bit value
- o 4: 8 bit map value
- o 5: 16 bit map value
- o 6: 32 bit map value
- o 7: 64 bit map value

Values 8 to 13 are reserved. Values 14 and 15 are experimental and may be specified locally in a segment routing domain.

The IPv6 identifier provides the low order 64 bits of IPv6 address. The high order 64 bit prefix is assumed to be common for all destinations. A fully qualified IPv6 address is created by combing the upper 64 bit prefix in the destination address of the packet with the identifier value as the low order 64 bits. The identifier type can be considered of form of compressing IPv6 addresses in segment identifiers when all the segment identifiers share the same prefix (.e.g a sixty-four bit prefix is common for all addresses in a segment routing domain.

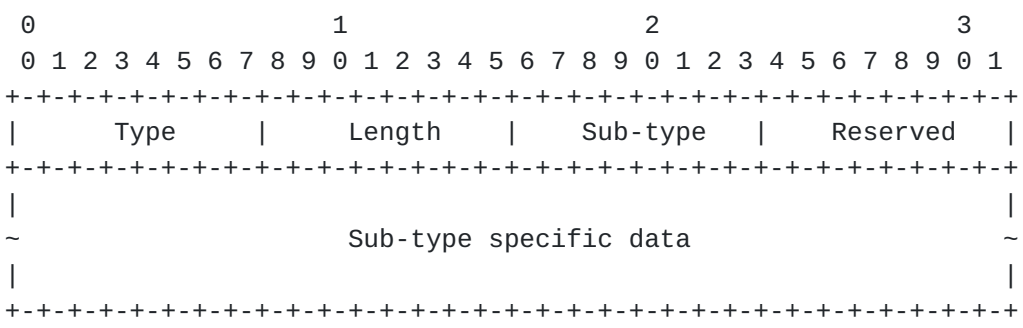
The IPv6 locator provides the high order 64 bits of IPv6 address. The low order 64 bits are assumed to be common for all destinations in a segment routing list. The use case for this would be Identifier/Locator protocols such as ILA or ILNP. A fully qualified address is created by combining the 64 bit prefix in the destination address of the packet with the identifier value as the low order 64 bits. The locatorr type can be considered of form of compressing IPv6 addresses in segment identifiers when all the segment identifiers share the same low order sixty-four bits.

Map values are mapped by receivers to fully qualified segment identifiers. A common use of this would be from receivers to maintain tables that map small segment identifiers to larger addresses. The table is specific within a segment routing domain must be managed accordingly. Using map values can be considerable savings in packet overhead when segment routing is used, particularly when a segment routing list would carry multiple IPv6 addresses.

## 3 Routing header security option

The routing header security options is a Destination Option that provides security for a following routing header.

The format of the option is:




Fields are:

- Type: Destination Option type for Routing Header security. This is a non-modifiable option and must not be ignored. Accordingly, the high order type bits are 010 to reflect that.

- Length: Variable length of option data.

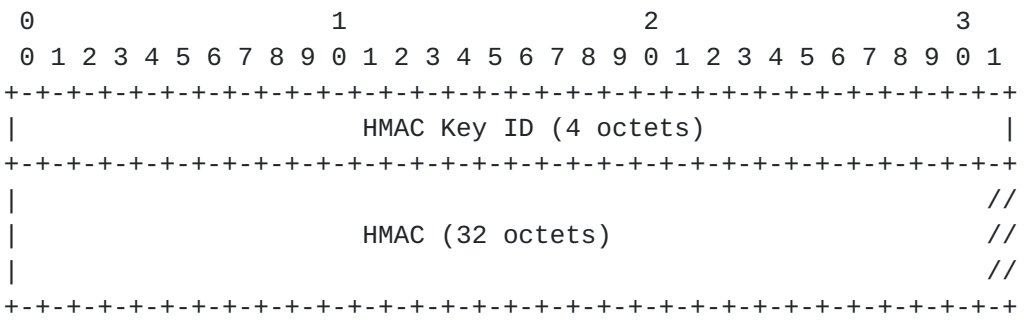
- Sub-type: Security method used. This specification defines one method of HMAC. See below.

- Reserved: MUST be set to zero when sending.

- Sub-type specific data: Data that is specific to the sub-type. The sub-type specific data for HMAC is described below.

### 3.1 HMAC Routing header security

HMAC Routing header security is a sub-type of routing header security. The format of the sub-type specific data is:




where:

- HMAC Key ID: 4 octets.
- HMAC: 32 octets.

The operation of the HMAC with respect routing header is specified in [SRV6EH].

### 3.2 Operation

The section describes the use and processing of the routing header security Destination Option.

#### 3.2.1 Sender operation

A sender may set a routing header security option in Destination Options that precede a Routing Header.

A sender SHOULD follow the recommended ordering in RFC8200 that a routing header immediately follows Destination Options that precede the routing header. As described below, if destination options immediately precede a routing header, a receive may apply the security option to the routing header while processing the security option as an optimization. To enable this, a sender MUST ensure that the Routing header immediately follows the Destination Options, and the routing header security option MUST be the last option in the Destination Options.

#### 3.2.2 Receiver operation

The security option MUST be processed by the last node in the routing header list of nodes to visit, and MAY be processed by intermediate

destinations. If a node does not process the option it MUST skip the option and proceed to the next option.

As demonstrated in the HMAC description, the routing header security option may be applied to fields in the routing header. Per [RFC8200], extension headers cannot be processed out of order so care must be taken to ensure processing order semantics are maintained. This specification presents two alternative methods that should yield the same effect.

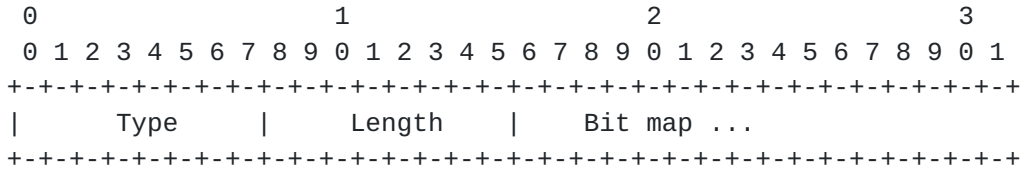
When a node processes a routing header security option in Destination Options, it can record the option data and make a note that the security is to be applied to the routing header. Subsequently, when the routing header is processed, the node can perform security verification as the first step in processing the routing header.

An alternative is for a node to perform the security verification when processing the security option in the Destination Options. A receiver MUST only do this if the Routing header immediately follows the Destination Options header and the routing header security option is the last Destination option.

## 4 Process per segment Destination Option

A sender MAY indicate that certain destination options preceding a Routing header are applicable to certain segments. A new option is defined for this functionality. This option SHOULD only be used if Destination Options immediately precedes a Routing header.

The format of the option is:




- Type: Destination Option type for the process per segment Destination Options. This is a non-modifiable option and must not be ignored. Accordingly, the high order type bits are 010 to reflect that.

- Length: Variable length of the bit map in octets.

- Bit map: A variable length bit map that describes which nodes are to process the option.

A position in the bit map corresponds to a Segments Left value in the

Routing header. For instance, if bit 3 in the bit map is set, then the option is processed by the node when Segments Left is 3. A bit map can indicate that multiple nodes are to process the options. Bits beyond the length of the bit map are assumed be zero.

If the length of the bit map is zero (i.e. length of the option data is zero) then any following options are MUST be processed by all intermediate nodes.

### 4.1 Sender operation

A sender MAY set the process per segment Destination Option. The Destination Options SHOULD immediately precede a Routing Header. The sender MAY indicate in the bit map that multiple intermediate destinations must process the options that following the process per segment option. Any Destination options MAY follow the process per segment option. A sender MAY direct options at different sets of intermediate destination by setting the per segment Destination option for each set. Options that precede a per segment Destination option are expected to be processed normally by each destination.

### 4.2 Receiver operation

When a node receives a process per segment Destination option it performs the following processing:

1) Check if the Next Header in the Destination Options header indicates a routing header (i.e. Next Header is equal to 43). If the next header is not a Routing header then processing the Destination Options is complete. Any following options are ignored.

2) Extract the Segments Left value from the Routing header immediately following the Destination Options header.

3) Determine the value in the bit map of the process per Segment option corresponding to the Segments Left value.

   - If the bit map value is set (bit is one) then process any following options to the end of the options list or another process per Segment option encountered.

   - If the bit map value is unset (bit is zero) then skip any following options to the end of the options list or another process per Segment option encountered.

4) If another process per Segment option is encountered process it starting from step #3 above

## 5 Security Considerations

This document defines new Destination option that is use to provide security for a routing header.

## 6 IANA Considerations

IANA is requested to assign two Destination options types.

IANA is requested to create a sub-type registry for the routing header security Destination Option.

## 7 References

### 7.1 Normative References

### 7.2 Informative References

[SRV6EH] Filsfils, C., Previdi, S., Leddy, J., Matsushima, S., and d. daniel.voyer@bell.ca, "IPv6 Segment Routing Header (SRH)", draft-ietf-6man-segment-routing-header-16.

[SRCOMP] Bonica, R., Xu, X., Chen, G., Zhu, Y., and Yang, G., "The IPv6 Compressed Routing Header (CRH)", draft-bonica-6man-comp-rtg-hdr-01.

[SRENDOPT] Bonica, R., Xu, X., Chen, G., Zhu, Y., and Yang, G., "The IPv6 Segment Endpoint Option", draft-bonica-6man-seg-end-opt-02

Author's Address

Tom Herbert
Quantonium
Santa Clara, CA
USA

Email: tom@quantonium.net