

Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: January 10, 2014

T. Eckert, Ed.
A. Zamfir
A. Choukir
Cisco
July 09, 2013

Flow Metadata Signaling with RSVP
draft-zamfir-tsvwg-flow-metadata-rsvp-00

Abstract

This specification proposes RSVP protocol extensions for signaling flow metadata attributes.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on January 10, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	2
1.1.	Requirements Language	2
2.	Flow Metadata Object	2
2.1.	FLOW_METADATA Class	3
2.1.1.	Basic IPFIX FLOW_METADATA Object	3
2.1.2.	Enhanced Protocol Independent FLOW_METADATA Object	3
2.2.	Semantic of carrying the Metadata Object	3
2.3.	Processing by a Non-Metadata Capable RSVP Router	4
2.4.	Processing by a Metadata Capable RSVP Router	4
3.	References	5
3.1.	Normative References	5
3.2.	Informative References	5
	Authors' Addresses	6

[1.](#) Introduction

Flow Metadata attributes are information elements (attributes) that identify flow characteristics, such as the type of media carried by application flows (e.g. video), the service class, the application that originated the flow, and others. The description of the Flow Metadata technology and some of the attribute definitions can be found in [[I-D.eckert-intarea-flow-metadata-framework](#)]. The flow attributes can be signaled over the flow path and inspected by intermediate network nodes for the purpose of applying differentiated flow treatment or collect network analytics. This specification proposes the use of RSVP as signaling protocol to carry the Flow Metadata using a new RSVP object. Two C-Type values are proposed for this object to allow for two possible encodings.

[1.1.](#) Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#) [[RFC2119](#)].

[2.](#) Flow Metadata Object

This specification proposes a new RSVP object with Class-Num from the 0x11bbbbbb range. To support informational metadata attribute processing on the path to the receiver, the sender inserts the Metadata object into an IPv4 or IPv6 Path message (i.e. Path messages

with SESSION Class = 1 and SENDER_TEMPLATE Class = 11). The Metadata object SHOULD appear only once in the message.

The object definition is given in [Section 2.1](#) while the details of processing are covered in [Section 2.2](#)

[2.1.](#) FLOW_METADATA Class

FLOW_METADATA Class = 234

Two encodings are defined, both of which carry the same IPFIX registered attributes as defined in [\[I-D.eckert-intarea-flow-metadata-framework\]](#). The first encoding (Basic IPFIX FLOW_METADATA) has less flexibility and lower encoding efficiency. This version of the encoding is referenced here for legacy reasons. It does not support a range of options that the second one does, including the signaling of sender and receiver attributes, security elements, distinction of originator of the attributes and ease of extensibility.

[2.1.1.](#) Basic IPFIX FLOW_METADATA Object

Basic IPFIX FLOW_METADATA Object: Class = 234, C-Type = 1

- o The metadata attributes are encoded in IPFIX format, as described in [\[RFC5101\]](#), with the following restrictions when creating the object:
 - * Options Template Record MUST NOT be present
 - * One and only one Template Record MUST be present
 - * One and only one Data Record MUST be included
- o An intermediate node that supports this specification SHOULD ignore any Options Template Record. It SHOULD only decode and process the first occurring Template and Data Records.

[2.1.2.](#) Enhanced Protocol Independent FLOW_METADATA Object

Enhanced Protocol Independent FLOW_METADATA Object: Class = 234, C-Type = 2

- o The contents and encoding rules for this object are specified in [[I-D.eckert-intarea-flow-metadata-framework](#)] and [[I-D.choukir-tsvwg-flow-metadata-encoding](#)].

[2.2.](#) Semantic of carrying the Metadata Object

The Metadata Object included in the Path message carries attributes from the sender of the flow towards the receiver. In some cases, e.g. if the sender does not support the generation and signaling of Metadata attribute, these attributes may be inserted by a proxy along the path of the flow. Metadata RSVP nodes on path may modify the

metadata attributes for purpose of influencing policy toward the receiver.

The node that originates Metadata information in a Path message may do so for the sole purpose of signaling Metadata information. In this case, the SENDER_TSPEC objects fields (as defined by [[RFC2210](#)]) should be set to 0:

- o Token Bucket Rate [r]
- o Token Bucket Size [b]
- o Peak Data Rate [p]
- o Minimum Policed Unit [m]

If the Metadata object is inserted in a Path message used for IntServ service [[RFC2210](#)] reservation requests, then all the rules of RSVP reservation request apply and in addition any actions driven purely by the metadata attributes may equally take place.

While the Metadata Object may be included in a Resv message, the specific processing rules for this option is left for followup documents or future versions of this specification.

[2.3.](#) Processing by a Non-Metadata Capable RSVP Router

As described in [[RFC2205](#)], a node that does not understand the Metadata object, should ignore but forward it, unexamined and

unmodified. When received in Path or Resv messages, it should be saved with the corresponding state and forwarded in any refresh message resulting from that state.

[2.4.](#) Processing by a Metadata Capable RSVP Router

The Metadata object may be inserted by the data flow initiating endpoint or network nodes along the path. The means by which an implementation determines the content of the Metadata object is outside the scope of this document.

Intermediate nodes that support this specification, decode the Flow Metadata information as indicated by the C-Type field only when received in Path message. Depending on the attributes, local configuration and policies, the node may take some actions. The Metadata attribute semantics are described in [[I-D.eckert-intarea-flow-metadata-framework](#)]. The received Flow Metadata object is stored against the Path state. When a subsequent Path message is received with a modified Metadata object, the

intermediate node determines the attributes that have been removed, modified and/or added by comparing the old and new objects, and takes appropriate actions.

As a result of these actions, an intermediate node may add new attributes to the Metadata object received in the Path message and signal them downstream. It can also modify some of the attributes present in the Flow Metadata object. RSVP does not have any transport protocol specific restrictions and the exact set of attributes that can be inserted and modified by intermediate nodes is described in [[I-D.eckert-intarea-flow-metadata-framework](#)]. Depending on local policies, an intermediate node may also remove some of the attributes received in the Metadata object of a Path message before forwarding downstream.

An intermediate node that receives a Resv message with a Metadata Object SHOULD store the object against the state and forward it unexamined and unmodified.

[3.](#) References

[3.1.](#) Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2205] Braden, B., Zhang, L., Berson, S., Herzog, S., and S. Jamin, "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", [RFC 2205](#), September 1997.
- [RFC2210] Wroclawski, J., "The Use of RSVP with IETF Integrated Services", [RFC 2210](#), September 1997.
- [RFC5101] Claise, B., "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information", [RFC 5101](#), January 2008.

[3.2.](#) Informative References

- [I-D.choukir-tsvwg-flow-metadata-encoding]
Eckert, T., Zamfir, A., Choukir, A., and C. Eckel,
"Protocol Independent Encoding for Signaling Flow
Characteristics", [draft-choukir-tsvwg-flow-metadata-
encoding-00](#) (work in progress), July 2013.
- [I-D.eckert-intarea-flow-metadata-framework]
Eckert, T., Penno, R., Choukir, A., and C. Eckel, "A
Framework for Signaling Flow Characteristics between

Eckert, et al.

Expires January 10, 2014

[Page 5]

Internet-Draft

FMD-RSVP

July 2013

Applications and the Network", [draft-eckert-intarea-flow-
metadata-framework-00](#) (work in progress), July 2013.

Authors' Addresses

Toerless Eckert (editor)
Cisco Systems, Inc.
San Jose
US

Email: eckert@cisco.com

Anca Zamfir
Cisco Systems, Inc.

EPFL, Quartier de l'Innovation
Ecublens, Vaud 1015
Switzerland

Email: ancaz@cisco.com

Amine Choukir
Cisco Systems, Inc.
EPFL, Quartier de l'Innovation
Ecublens, Vaud 1015
CH

Phone: +41 78 75 98 561
Email: amchouki@cisco.com