

IDR Working Group  
Internet-Draft  
Updates: [RFC5575](#) (if approved)  
Intended status: Standards Track  
Expires: September 21, 2014

R. Raszuk, Ed.  
NTT MCL Inc.  
B. Pithawala  
Cisco Systems  
D. McPherson  
Verisign, Inc.  
A. Karch  
Cisco Systems  
March 20, 2014

**Dissemination of Flow Specification Rules for IPv6**  
**draft-ietf-idr-flow-spec-v6-05**

Abstract

Dissemination of Flow Specification Rules [[RFC5575](#)] provides a protocol extension for propagation of traffic flow information for the purpose of rate limiting or filtering. The [[RFC5575](#)] specifies those extensions for IPv4 protocol data packets.

This specification extends the current [[RFC5575](#)] and defines changes to the original document in order to make it also usable and applicable to IPv6 data packets.

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 September 21, 2014.

Copyright Notice

Copyright (c) 2014 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

<a href="#">1.</a>	Introduction . . . . .	<a href="#">2</a>
<a href="#">2.</a>	IPv6 Flow Specification encoding in BGP . . . . .	<a href="#">2</a>
<a href="#">3.</a>	IPv6 Flow Specification types changes . . . . .	<a href="#">3</a>
<a href="#">3.1.</a>	Order of Traffic Filtering Rules . . . . .	<a href="#">5</a>
<a href="#">4.</a>	IPv6 Flow Specification Traffic Filtering Action changes . .	<a href="#">6</a>
<a href="#">5.</a>	Security considerations . . . . .	<a href="#">7</a>
<a href="#">6.</a>	IANA Considerations . . . . .	<a href="#">7</a>
<a href="#">7.</a>	Acknowledgments . . . . .	<a href="#">7</a>
<a href="#">8.</a>	References . . . . .	<a href="#">8</a>
<a href="#">8.1.</a>	Normative References . . . . .	<a href="#">8</a>
<a href="#">8.2.</a>	Informative References . . . . .	<a href="#">8</a>
	Authors' Addresses . . . . .	<a href="#">8</a>

## [1.](#) Introduction

The growing amount of IPv6 traffic in private and public networks requires the extension of tools used in the IPv4 only networks to be also capable of supporting IPv6 data packets.

In this document authors analyze the differences of IPv6 [\[RFC2460\]](#) flows description from those of traditional IPv4 packets and propose subset of new encoding formats to enable Dissemination of Flow Specification Rules [\[RFC5575\]](#) for IPv6.

This specification should be treated as an extension of base [\[RFC5575\]](#) specification and not its replacement. It only defines the delta changes required to support IPv6 while all other definitions and operation mechanisms of Dissemination of Flow Specification Rules will remain in the main specification and will not be repeated here.

## [2.](#) IPv6 Flow Specification encoding in BGP

The [\[RFC5575\]](#) defines a new SAFIs (133 for IPv4) and (134 for VPNv4) applications in order to carry corresponding to each such application flow specification.



This document will redefine the [[RFC5575](#)] SAFIs in order to make them AFI specific and applicable to both IPv4 and IPv6 applications.

The following changes are defined:

"SAFI 133 for IPv4 dissemination of flow specification rules" to now be defined as "SAFI 133 for dissemination of unicast flow specification rules"

"SAFI 134 for VPNv4 dissemination of flow specification rules" to now be defined as "SAFI 134 for dissemination of L3VPN flow specification rules"

For both SAFIs the indication to which address family they are referring to will be recognized by AFI value (AFI=1 for IPv4 or VPNv4, AFI=2 for IPv6 and VPNv6 respectively). Such modification is fully backwards compatible with existing implementation and production deployments.

It needs to be observed that such choice of proposed encoding is compatible with filter validation against routing reachability information as described in [section 6 of RFC5575](#). Validation tables will now be performed according to the following rules.

Flow specification received over AFI/SAFI=1/133 will be validated against routing reachability received over AFI/SAFI=1/1

Flow specification received over AFI/SAFI=1/134 will be validated against routing reachability received over AFI/SAFI=1/128

Flow specification received over AFI/SAFI=2/133 will be validated against routing reachability received over AFI/SAFI=2/1

Flow specification received over AFI/SAFI=2/134 will be validated against routing reachability received over AFI/SAFI=2/128

### **[3.](#) IPv6 Flow Specification types changes**

The following component types are redefined or added for the purpose of accommodating new IPv6 header encoding. Unless otherwise stated all other types as defined in [RFC5575](#) apply to IPv6 packets as is.

Type 1 - Destination IPv6 Prefix

Encoding: <type (1 octet), prefix offset (1 octet), prefix length (1 octet), prefix>



Defines the destination prefix to match. Prefix offset has been defined to allow for flexible matching on part of the IPv6 address where we want to skip (don't care) of N first bits of the address. This can be especially useful where part of the IPv6 address consists of an embedded IPv4 address and matching needs to happen only on the embedded IPv4 address. The encoded prefix contains enough octets for the bits used in matching (length minus offset bits).

#### Type 2 - Source IPv6 Prefix

Encoding: <type (1 octet), prefix offset (1 octet), prefix length (1 octet), prefix>

Defines the source prefix to match. Prefix offset has been defined to allow for flexible matching on part of the IPv6 address where we want to skip (don't care) of N first bits of the address. This can be especially useful where part of the IPv6 address consists of an embedded IPv4 address and matching needs to happen only on the embedded IPv4 address. The encoded prefix contains enough octets for the bits used in matching (length minus offset bits).

#### Type 3 - Next Header

Encoding: <type (1 octet), [op, value]+>

Contains a set of {operator, value} pairs that are used to match the last Next Header value octet in IPv6 packets. The operator byte is encoded as specified in component type 3 of [[RFC5575](#)].

While IPv6 allows for more than one Next Header field in the packet the main goal of Type 3 flow specification component is to match on the subsequent IP protocol value. Therefore the definition is limited to match only on last Next Header field in the packet.

#### Type 12 - Fragment

Encoding: <type (1 octet), [op, bitmask]+>

Uses bitmask operand format defined above. Bit-7 is not used and MUST be 0 to provide backwards-compatibility with the definition in [RFC5575](#).



```

      0   1   2   3   4   5   6   7
+---+---+---+---+---+---+---+---+
|   Reserved   |LF|FF|IsF| 0 |
+---+---+---+---+---+---+---+

```

Bitmask values:

+ Bit 6 - Is a fragment (IsF)

+ Bit 5 - First fragment (FF)

+ Bit 4 - Last fragment (LF)

Type 13 - Flow Label - New type

Encoding: <type (1 octet), [op, value]+>

Contains a set of {operator, value} pairs that are used to match the 20-bit Flow Label field [[RFC2460](#)]. The operator byte is encoded as specified in the component type 3 of [[RFC5575](#)].

The following example demonstrates the new prefix encoding for: "all packets to ::1234:5678:9A00:0/80-104 from 192::/8 and port {range [137, 139] or 8080}". In the destination prefix, "80-" represents the prefix offset of 80 bits. In this example, the 0 offset is omitted from the printed source prefix.

```

+-----+-----+-----+
| destination           | source       | port         |
+-----+-----+-----+
| 0x01 40 68 12 34 56 78 9A | 02 00 08 c0 | 04 03 89 45 8b 91 1f 90 |
+-----+-----+-----+

```

### [3.1.](#) Order of Traffic Filtering Rules

The original definition for the order of traffic filtering rules can be reused with new consideration for the IPv6 prefix offset. As long as the offsets are equal, the comparison is the same, retaining longest-prefix-match semantics. If the offsets are not equal, the lowest offset has precedence, as this flow matches the most significant bit.

Pseudocode:

```

flow_rule_v6_cmp (a, b)
{

```





```
comp1 = next_component(a);
comp2 = next_component(b);
while (comp1 || comp2) {
    // component_type returns infinity on end-of-list
    if (component_type(comp1) < component_type(comp2)) {
        return A_HAS_PRECEDENCE;
    }
    if (component_type(comp1) > component_type(comp2)) {
        return B_HAS_PRECEDENCE;
    }

    if (component_type(comp1) == IPV6_DESTINATION || IPV6_SOURCE) {
        // offset not equal, lowest offset has precedence
        // offset equal ...
        common_len = MIN(prefix_length(comp1), prefix_length(comp2));
        cmp = prefix_compare(comp1, comp2, offset, common_len);
        // not equal, lowest value has precedence
        // equal, longest match has precedence
    } else {
        common =
            MIN(component_length(comp1), component_length(comp2));
        cmp = memcmp(data(comp1), data(comp2), common);
        // not equal, lowest value has precedence
        // equal, longest string has precedence
    }
}

return EQUAL;
}
```

#### **4. IPv6 Flow Specification Traffic Filtering Action changes**

One of the traffic filtering actions which can be expressed by BGP extended community is defined in [\[RFC5575\]](#) as traffic-marking. This extended community type is of value: 0x8009.

For the purpose of making it compatible with IPv6 header action expressed by presence of this extended community has been modified to read:

Traffic Marking: The traffic marking extended community instructs a system to modify first 6 bits of Traffic Class field as (recommended by [\[RFC2474\]](#)) of a transiting IPv6 packet to the corresponding value. This extended community is encoded as a sequence of 42 zero bits followed by the 6 bits overwriting DSCP portion of Traffic Class value.



Another traffic filtering action defined in [[RFC5575](#)] as a BGP extended community is redirect. To allow an IPv6 address specific route-target, a new traffic action IPv6 address specific extended community is provided. The extended community type has the value 0x800b.

Redirect-IPv6: The redirect IPv6 address specific extended community allows the traffic to be redirected to a VRF routing instance that lists the specified IPv6 address specific route-target in its import policy. If several local instances match this criteria, the choice between them is a local matter (for example, the instance with the lowest Route Distinguisher value can be elected). This extended community uses the same encoding as the IPv6 address specific Route Target extended community [[RFC5701](#)].

## **5. Security considerations**

No new security issues are introduced to the BGP protocol by this specification.

## **6. IANA Considerations**

IANA is requested to rename currently defined SAFI 133 and SAFI 134 per [[RFC5575](#)] to read:

133	Dissemination of flow specification rules
134	L3VPN dissemination of flow specification rules

IANA is requested to create and maintain a new registry entitled: "Flow Spec IPv6 Component Types". The following component types have been registered:

Type 1	- Destination IPv6 Prefix
Type 2	- Source IPv6 Prefix
Type 3	- Next Header
Type 4	- Port
Type 5	- Destination port
Type 6	- Source port
Type 7	- ICMP type
Type 8	- ICMP code
Type 9	- TCP flags
Type 10	- Packet length
Type 11	- DSCP
Type 12	- Fragment
Type 13	- Flow Label

## **7. Acknowledgments**



Authors would like to thank Pedro Marques, Hannes Gredler and Bruno Rijnsman, Brian Carpenter, and Thomas Mangin for their valuable input.

## **8. References**

### **8.1. Normative References**

- [I-D.ietf-6man-flow-3697bis]  
Amante, S., Carpenter, B., Jiang, S., and J. Rajahalme,  
"IPv6 Flow Label Specification", [draft-ietf-6man-flow-3697bis-07](#) (work in progress), July 2011.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", [RFC 2460](#), December 1998.
- [RFC2474] Nichols, K., Blake, S., Baker, F., and D. Black,  
"Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", [RFC 2474](#), December 1998.
- [RFC4271] Rekhter, Y., Li, T., and S. Hares, "A Border Gateway Protocol 4 (BGP-4)", [RFC 4271](#), January 2006.
- [RFC5492] Scudder, J. and R. Chandra, "Capabilities Advertisement with BGP-4", [RFC 5492](#), February 2009.
- [RFC5575] Marques, P., Sheth, N., Raszuk, R., Greene, B., Mauch, J., and D. McPherson, "Dissemination of Flow Specification Rules", [RFC 5575](#), August 2009.
- [RFC5701] Rekhter, Y., "IPv6 Address Specific BGP Extended Community Attribute", [RFC 5701](#), November 2009.

### **8.2. Informative References**

- [RFC5095] Abley, J., Savola, P., and G. Neville-Neil, "Deprecation of Type 0 Routing Headers in IPv6", [RFC 5095](#), December 2007.

Authors' Addresses



Robert Raszuk (editor)  
NTT MCL Inc.  
101 S Ellsworth Avenue Suite 350  
San Mateo, CA 94401  
US

Email: [robert@raszuk.net](mailto:robert@raszuk.net)

Burjiz Pithawala  
Cisco Systems  
170 West Tasman Drive  
San Jose, CA 95134  
US

Email: [bpithaw@cisco.com](mailto:bpithaw@cisco.com)

Danny McPherson  
Verisign, Inc.

Email: [dmcpherson@verisign.com](mailto:dmcpherson@verisign.com)

Andy Karch  
Cisco Systems  
170 West Tasman Drive  
San Jose, CA 95134  
US

Email: [akarch@cisco.com](mailto:akarch@cisco.com)



