LSR Working Group
Louis Chan
INTERNET-DRAFT Krzysztof Szarkowicz
Intended status: Standard Track Juniper Networks
Gyan Mishra

Verizon Inc.

Expires: Jan 7, 2024 Jul 7, 2023

IGP extensions for Advertising Offset for Flex-Algorithm
draft-chan-lsr-igp-adv-offset-03.txt

Abstract

This document describes the IGP extensions to provide predictable Adjacency-SIDs per Flex-Algorithm [FLEXALGO] in segment routing.

We propose some methods to allow the advertisement of additional TLV in IGP so that the Flex-Algorithm specific Adjacency-SIDs could be automatically derived.

With the proposed method, the size of advertisement on per node per link basis is greatly reduced. Each participating router would derive the required labels automatically.

Extensions for offset to derive Flex-Algorithm Prefix-SID is also included in the document.

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 Jan 7, 2024.

Copyright Notice

Table of Contents

1. Introduction

The draft proposes methods for routers to announce Flex-Algorithm specific
Adjacency-SID with minimal advertisement in IGP. When the other routers need a SR
policy (aka SR-TE) or TI-LFA path, the Flex-Algorithm specific Adjacency-SID would
be taken into the path label construction.

Hence, the top most label, either node or link related, in a SR policy stack would be used to identify a certain Flex-Algo with full identification.

In the same draft, Virtual Flex-Algorithm (VFA) concept is introduced.

2. Conventions used in this document

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].

In this document, these words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying significance described in RFC 2119.

3. Virtual Flex-Algorithm

Here in this draft, Virtual Flex-Alogrithm (VFA) is introduced. There could be additional virtual topology which would like to share the same Flex-Algo topology with same metrics and constraints, and VFA would need different Prefix-SID and Adj-SID for identification in the forwarding plane.

It would save resource to recalculate the shared topology for different VFA.

The relationship between VFA and its related Flex-Algorithm could be similar to VLAN and port. Or, they could be in parallel relationship.

VFA would be represented in 32 bits with the minimum value of 256.

There are different VFA use cases that could be applied as described below. FA, here means Flex-Algo.

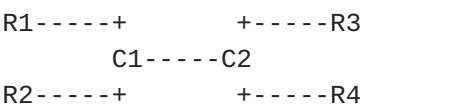
If the network nodes shared the same SRGB range, for Prefix-SID MPLS label, it

would looks like this

[FA-ID][node] or [VFA-ID][node]

The FA portion or VFA portion of MPLS label would be the identification of QOS across network. Since SRGB is the same, the values of FA-ID and VFA-ID are consistent.

3.1. Multi-point to Multi-point Hierarchical QOS



In this diagram, this is a multipoint VPN application.

R1 could be on VFA600, and R2 on VFA601.
Both VFA600 and VFA601 is based on FA129 (Flex-Algo 129).

FA129 is the parent of VFA600 and VFA601 in terms of hierarchical QOS (H-QOS)
Router C2 could apply H-QOS when sending to R3 for traffic from R1 and R2.

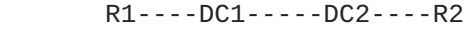
e.g. FA129 10Gbps in total
VFA600 6Gbps/4Gbps for PIR/CIR
VFA601 8Gbps/4Gbps for PIR/CIR

This resembles some operations like VP shaping and VC shaping in ATM days.

This draft only deals with how to label the packets, and how the QOS is enforced
and signaled is out of the current scope.

Here in this example, there are only two levels of hierarchy. It could have
multiple levels with more VFA. e.g. geographical region levels. And, FA129 is used
in this example, but in fact, Algo 0 could also be used with VFA concept.

3.2. DC interconnect



DC1 and DC2 are data center gateway routers.

Between R1 and R2, it could run multiple VFA, like VFA600 and VFA601

But between DC1 and DC2, it only runs FA129.

If using compatibility mode operation (described in section 8.3), VFA600 and VFA601
packets are tunneled through FA129 between DC1 and DC2.

Transport labels are stacked in this case. E.g. Top transport label is from FA129,
and the next transport label is from VFA600.

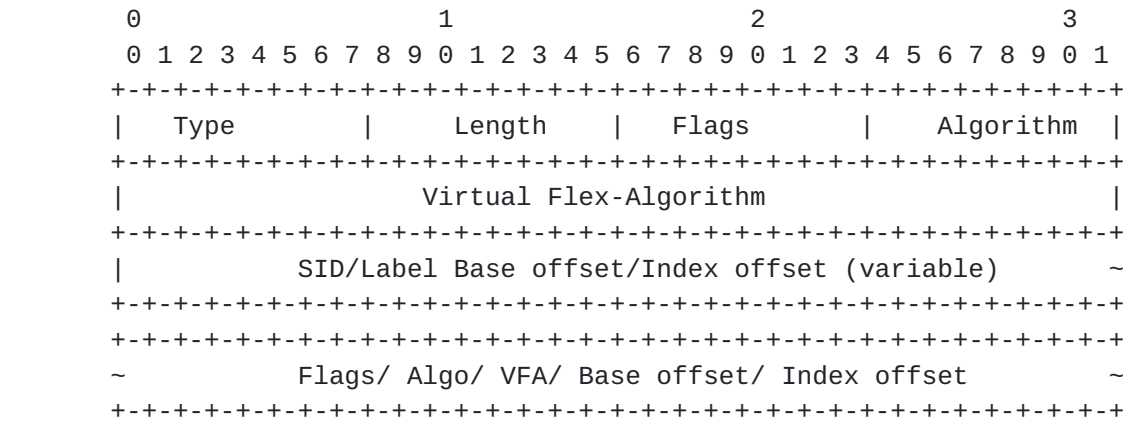
At R1 and R2, individual QOS could be applied to VFA600 or VFA601 packets with the
corresponding labels. VFA600 is a 2Gbps service, and VFA601 is a 4Gbps service.
Both are low latency services within FA129.

The application is similar in using VLAN trunk port and VLANs.

4. ISIS extension

With reference to RFC8667, the information could be advertised in Router Capabilities as sub-TLV in Section 3.

4.1. Algorithm Offset for Adj-SID

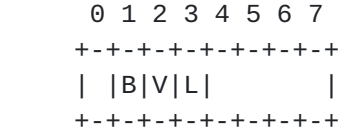


where:

Type: TDB

Length: variable; multiple advertisements

Flags: 1 octet field of the following flags:



where:

B-Flag: Backup Flag. If set, the Adj-SID is eligible for protection (e.g., using IP Fast Reroute (IPFRR) or MPLS Fast Reroute (MPLS-FRR)) as described in [RFC8402].

V-Flag: Value Flag. If set, then it carries a label value. If not set, it is an index value.

L-Flag: Local Flag. Always set to 1

Algorithm: 1 octet
Algorithm or Flex-Algo value is either 0 or between 128 and
255.

Virtual Flex-Algorithm: 4 octets
Value 0 means that it is not VFA
Value >=256 means VFA identification number, and the base
topology is denoted by Algorithm
Value 1 to 127 are reserved
Value 128 to 255 are invalid

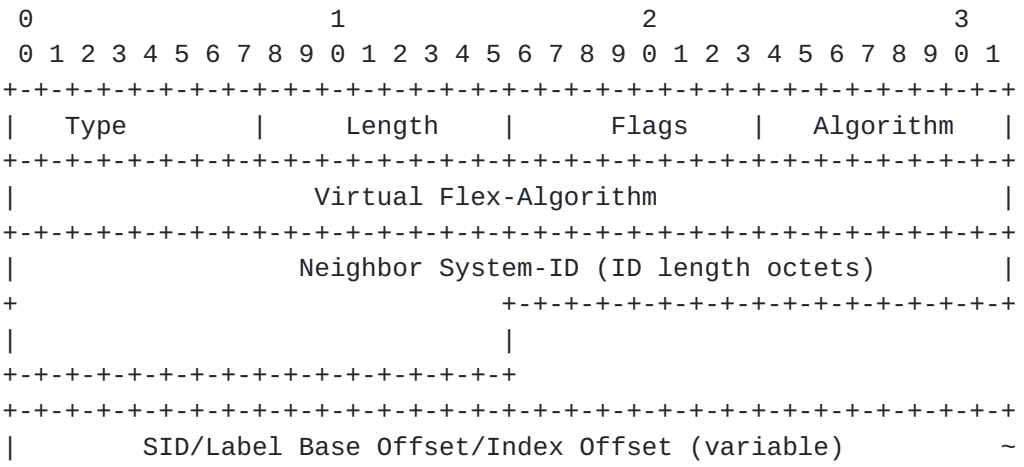
Base Offset: 3 octets
Local label base for Adjacency-SIDs for given Flex-
Algorithm. The derived Adjacency-SID is the sum of the base
offset and the Algo 0 label

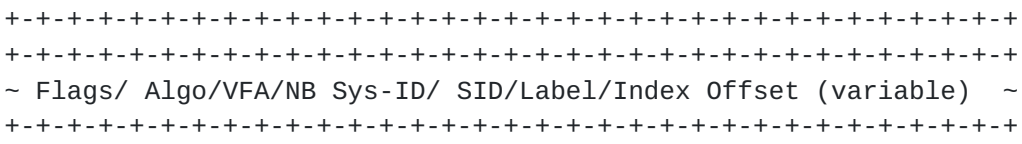
Index Offset: 4 octets
If Adjacency-SID is advertised as index, this provides an
offset index value. The new label is sum of (SRGB base +
index + index offset)

The format of Base Offset and Index Offset is the same as RFC8667. The choice of advertising Base Offset or Index Offset must match the advertisement of original Adjacency-SID method from the same router.

Flags F, S and P are inherited from individual Adj-sid advertisement of Algo 0

4.2. Algorithm Offset for LAN based Adj-SID



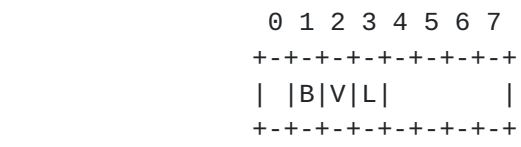


where:

Type: TDB

Length: variable; multiple advertisements

Flags: 1 octet field of the following flags:



where:

B-Flag: Backup Flag. If set, the Adj-SID is eligible for protection (e.g., using IP Fast Reroute (IPFRR) or MPLS Fast Reroute (MPLS-FRR)) as described in [RFC8402].

V-Flag: Value Flag. If set, then it carries a label value. If not set, it is an index value.

L-Flag: Local Flag. Always set to 1

Algorithm: 1 octet. Same as Section 4.1

Virtual Flex-Algorithm: 4 octets. Same as Section 4.1

Neighbor System-ID: IS-IS System-ID of length "ID Length" as defined in [ISO10589].

Base Offset: 3 octets. Same as Section 4.1

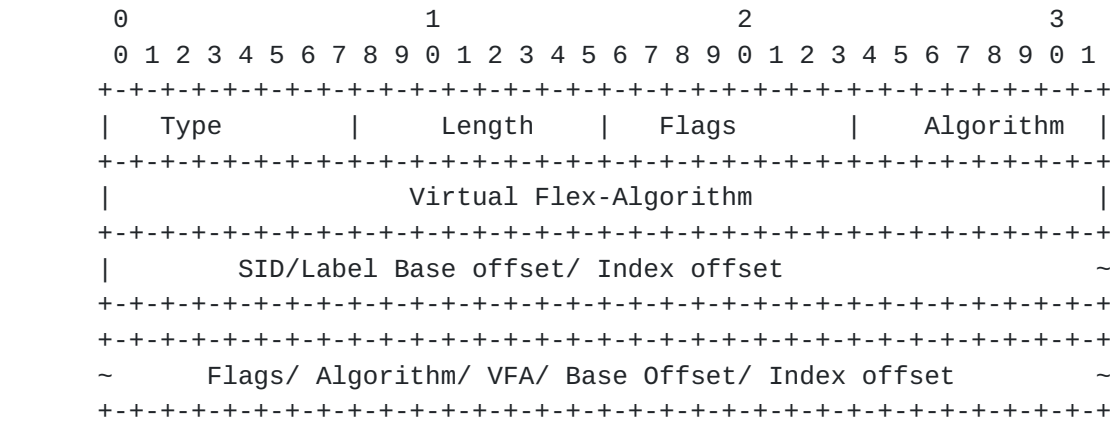
Index Offset: 4 octets. Same as Section 4.1

Other flags: inherited from Algo 0. See Section 4.1

4.3. Algorithm Offset for Prefix SID

Prefix-SID for Algorithm could also be generated by adding an index offset value or
a base offset for label.

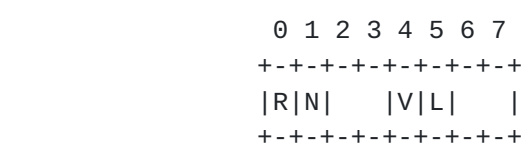
This advertisement is an sub-TLV in RFC8667 Section 2.



Type: TDB

Length: variable

Flags: 1 octet field of the following flags:



where:

R-Flag: Re-advertisement Flag. If set, then the prefix to which this Prefix-SID is attached has been propagated by the router from either another level (i.e., from Level-1 to Level-2 or the opposite) or redistribution (e.g., from another protocol).

N-Flag: Node-SID Flag. If set, then the Prefix-SID refers to the router identified by the prefix. Typically, the N-Flag is set on Prefix-SIDs that are attached to a router loopback address. The N-Flag is set when the Prefix-SID is a Node-SID as described in [RFC8402].

V-Flag: Value Flag. If set, then it carries a label value. If not set, it is an index value.

L-Flag: Local Flag. Always set to 0

Algorithm:	1 octet Algorithm/Flex-Algo value is either 0 or between 128 and 255

Virtual Flex-Algorithm:	4 octets Value 0 means that it is not VFA Value >=256 means VFA identification number, and the base topology is denoted by Algorithm Value 1 to 127 is reserved Value 128 to 255 are invalid
Base offset:	3 octets Label base for Prefix-SIDs for given Algorithm The derived Prefix-SID is the sum of the base Offset and the Algo 0 label
Index Offset:	4 octets Index offset counting from Algo 0 The new label for Prefix is sum of (SRGB base + index + index offset)

The format of Base offset and Index Offset is the same as RFC8667. The choice of advertising Base Offset or Index Offset MUST match the advertisement of original Prefix-SID method from the same router.

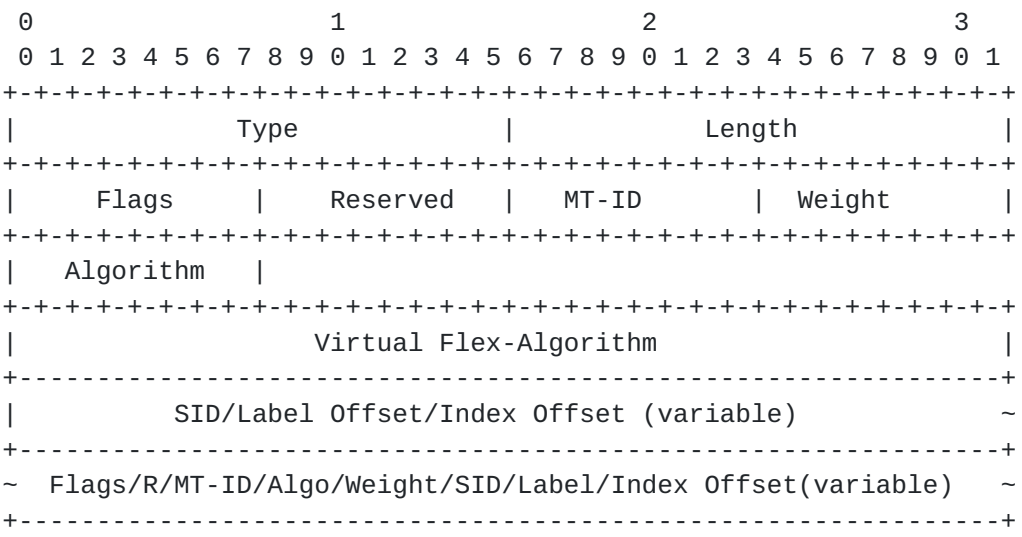
Flags P and E are inherited from Algo 0 advertisement. Use of Flag E might need some investigation.

Flags R and N are under discussion to be removed from this draft.

5. OSPF extension

The following TLVs are advertised in the Router Information Opaque LSA (defined in [RFC7770]). These TLVs are applicable to both OSPFv2 and OSPFv3.

5.1. Algorithm Offset for Adj-SID

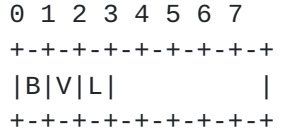


where:

Type: TBD

Length: Variable

Flags: Single-octet field containing the following flags:



where:

B-Flag: Backup Flag. If set, the Adj-SID is eligible for protection (e.g., using IP Fast Reroute (IPFRR) or MPLS Fast Reroute (MPLS-FRR)) as described in [RFC8402].

V-Flag: Value Flag. If set, then it carries a label value. If not set, it is an index value.

L-Flag: Local Flag. Always set to 1

Other bits: Reserved. These MUST be zero when sent and are ignored when received.

MT-ID: Multi-Topology ID (as defined in [RFC4915]

Weight: Weight used for load-balancing purposes. The use of the weight is defined in [RFC8402].

Algorithm: 1 octets
Algorithm/Flex-Algo value is either 0 or between 128 and 255.

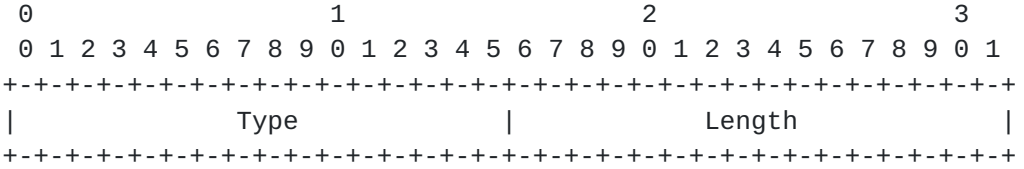
Virtual Flex-Algorithm: 4 octets
Value 0 means that it is not VFA
Value >=256 means VFA identification number, and the base topology is denoted by Algorithm
Value 1 to 127 is reserved
Value 128 to 255 are invalid

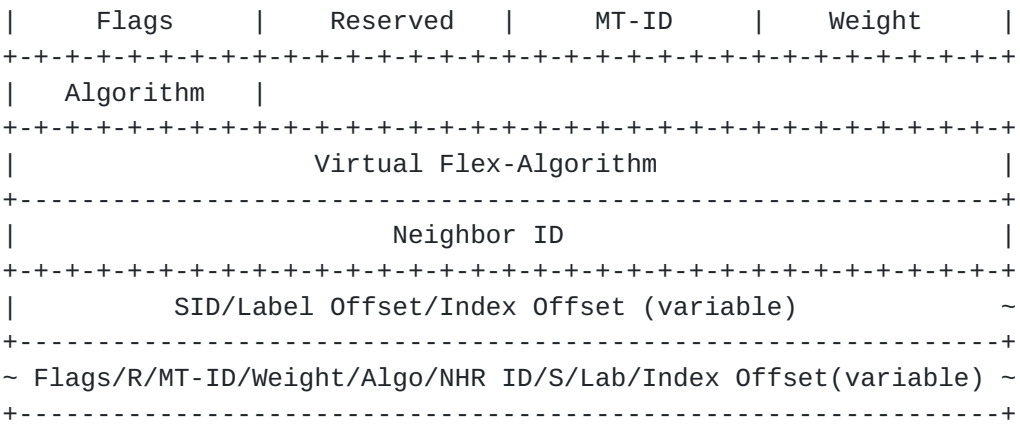
Base Offset: 3 octets
Local label base for Adjacency-SIDs for given Flex-Algorithm. The derived Adjacency-SID is the sum of the base offset and the Algo 0 label

Index Offset: 4 octets
If Adjacency-SID is advertised as index, this provides an offset index value. The new label is sum of (SRGB base + index + index offset)

The format of Base Offset and Index Offset is the same as RFC8665 or RFC8666. The choice of advertising Base Offset or Index Offset must match the advertisement of original Adjacency-SID method from the same router.

5.2. Algorithm Offset for LAN based Adj-SID

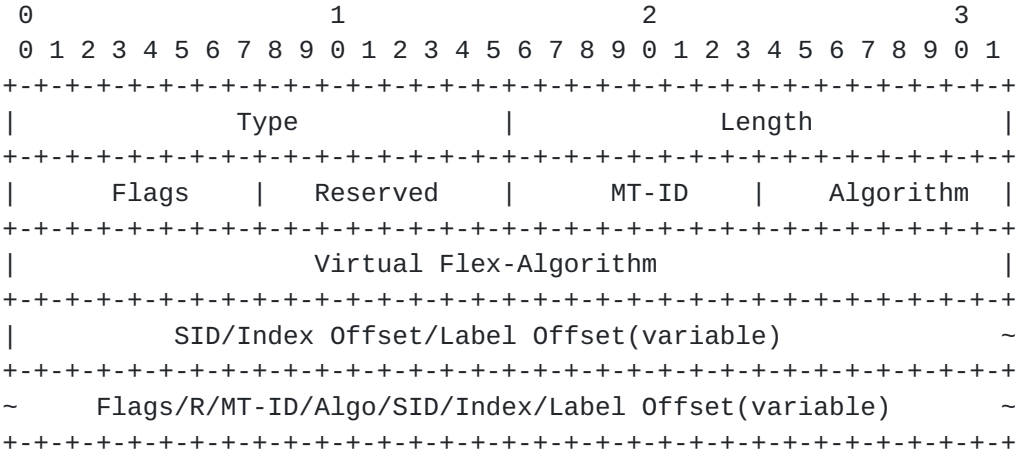



| Flags | Reserved | MT-ID | Weight |
| Algorithm |
| Virtual Flex-Algorithm |
| Neighbor ID |
| SID/Label Offset/Index Offset (variable) ~
~ Flags/R/MT-ID/Weight/Algo/NHR ID/S/Lab/Index Offset(variable) ~

where:

Type:	TBD
Length:	Variable
Flags:	Same as 5.1
Reserved:	Same as 5.1
MT-ID:	Same as 5.1
Weight:	Same as 5.1
Algorithm:	Same as 5.1
Virtual Flex-Algorithm:	Same as 5.1
Neighbor ID:	The Router ID of the neighbor for which the LAN Adjacency SID is advertised
Base offset:	Same as 5.1
Index Offset:	Same as 5.1

5.3. Algorithm Offset for Prefix SID



where:

Type:	TDB

Length: Variable

Flags: Single-octet field. The following flags are defined:

```
 0  1  2  3  4  5  6  7
+--+--+--+--+--+--+--+--+
|  |  |  |E |V |L |  |  |
+--+--+--+--+--+--+--+--+
```

E-Flag: Explicit Null Flag. If set, any upstream neighbor of the Prefix-SID originator MUST replace the Prefix-SID with the Explicit NULL label (0 for IPv4) before forwarding the packet.

V-Flag: Value Flag. If set, then it carries a label value. If not set, it is an index value.

L-Flag: Local Flag. Always set to 0

Other bits: Reserved. These MUST be zero when sent and are ignored when received.

MT-ID: Multi-Topology ID (as defined in [RFC4915])

Algorithm: 1 octet
Algorithm/Flex-Algo value is either 0 or between 128 and 255

Virtual Flex-Algorithm: 4 octets
Value 0 means that it is not VFA
Value >=256 means VFA identification number, and the base topology is denoted by Algorithm
Value 1 to 127 is reserved
Value 128 to 255 are invalid

Base offset: 3 octets
Label base for Prefix-SIDs for given Algorithm
The derived Prefix-SID is the sum of the base Offset and the Algo 0 label

Index Offset: 4 octets
Index offset counting from Algo 0
The new label for Prefix is sum of (SRGB base + index + index offset)

The format of Base offset and Index Offset is the same as RFC8665 or RFC8666. The choice of advertising Base Offset or Index Offset MUST match the

advertisement of
      original Prefix-SID method from the same router.

6. Allowed configurations

	FA	FA	FA	VFA	VFA	VFA
Adj-SID offset	P	P	N	P	P	N
Prefix-sid offset	N	P	N	N	P	P
Valid combination	Y	N	Y	N	Y	Y
Note			#1			#2

P means the sub-tlv is present

N means the sub-tlv is not present

#1 - For this Flex-Algo, it will use the adj-sid from Algo 0

#2 - For this VFA, it will use the Adj-sid from base algorithm, which could be either Algo 0 or the associated Flex-Algo. Only if the base Flex-Algo has no advertisement of Adj-sid offset, it then would use Adj-sid of Algo 0 as fallback mechanism.

7. Example for illustration

One node advertises the following information in IGP.

Algo 0 Prefix-sid label: 400001

Adj-sid labels for 99 interfaces: 32, 33, 34...130

SRGB starts with 400000

Value	Algo 0	FA128	VFA500	FA129	VFA600	VFA601	VFA602
Base Algo	n/a	n/a	128	n/a	129	129	129
Adj-sid offset	n/a	NIL	NIL	2000	6000	7000	NIL
Pre-sid offset	n/a	n/a	5000	n/a	6000	7000	

```
8000  |
        |Prefix-sid (N)| 400001  | 401001| 405001| 402001| 406001| 407001|
408001|
        |Intf#1        |   32    |  32   |  32   |  2032 |  6032 | 7032  |
2032 |
        |Intf#2        |   33    |  n/a  |  n/a  |  2033 |  6033 | 7033  |
2033 |
        |Intf#3        |   34    |  34   |  34   |  n/a  |   n/a |  n/a  |  n/
a  |
        |.....         |   ..    |  ..   |  ..   |  ..   |   ..  |  ..
|  ..   |
        |Intf#99       |   130   |  130  |  130  |  2130 |  6130 | 7130  |
2130 |
        |Note          |         |  #1   |   #2  |   #3  |  #4   |       |
#5   |

+-------------------------------------------------------------------------+
```

Note:

#1 There is no Adj-sid offset advertised for Flex-Algo 128. Hence FA128 shares Adj-sid from Algo 0.

#2 Since FA128 has no specific Adj-sid, VFA500 follows Adj-sid from Algo 0.

#3 There is only Adj-sid offset advertised for Flex-Algo 129. The Adj-sid would be added with 2000 offset. E.g. 32 becomes 2032.
Please note that there is no Adj-sid for Intf#3 in FA129.

#4 Both Adj-sid offset and Prefix-sid offset are advertised for VFA600. Prefix-sid of 406001 (400001 + 6000) and Adj-sid of 6032 to 6130 are used.

#5 VFA602 uses Adj-sid's from FA129.

NIL means the TLV/sub-TLV is not present.

The QOS relationship between FA129, VFA600 and VFA601 could be in multiple forms.
e.g.
a. FA129 is the parent of VFA600 and VFA601. It is similar to stacked vlan.
b. FA129, VFA600 and VFA601 are all in the same level. It is similar to parallel vlans.
c. FA129 is parent of VFA600, but in parallel with VFA601. It is a mix in QOS relationship.

In this draft, it only describes how the labels are assigned, and the actual QOS enforcement is out of the scope.

For FA129, when the numbering system is carefully designed, labels such as, 402xxx for network wide, could be used to identify FA129, and apply QOS along the transit nodes. Label of 2xxx is for local node consumption when QOS is applied.

8. Compatibility

8.1. Legacy nodes which support only Flex-Algo

For nodes that do not support Adj-sid offset, the label stack could use Adj-SID
from Algo 0, providing that the node still support Flex-Algo Prefix SID.

Since it cannot understand Prefix-sid offset, it will not participate in any VFA.

8.2. TI-LFA calculation

For TI-LFA, there would be two modes of operation, loose or strict.

For strict mode, all Adj-SID involved in TI-LFA candidate path must be derived with
offset method, when each node in the path has announced such Adj-SID offset.

For loose mode, Adj-SID in the TI-LFA candidate path could be a mix of Adj-SID's
with or without offset. This allows backward compatibility with routers which only
support Flex-Algo.

Default is loose mode.

8.3. Backward compatibility for VFA

There would be a few options for backward compatibility.

All routers below support the new extensions in the draft, except R3. R3 is only
capable of running Flex-Algo.

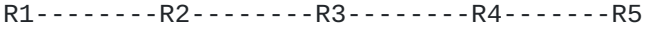
Here we assume a case of one FA plus one VFA.

Node-sid					
FA129	402001	402002	402003	402004	402005
VFA600	406001	406002	n/a	406004	406005

For a packet based on VFA600 from R1 to R5, the top label would be encoded in
one of the following fallback options.

8.3.1. Option 1: Fallback only to base Flex-Algo

Fallback to FA129 in this example



top label	406005	402005	402005	402005(or popped)

R2 detects R3 could not support VFA600, and it allows VFA600 to fallback FA129
node-sid for R5. It is 402005 here.

R2 is the PLR.

8.3.2. Option 2: Fallback with tunnel

It is similar to TI-LFA.

```
                R1--------R2--------R3--------R4-------R5

     top labels     406005   402004     406005    406005(or popped)
                             406005
```

R2 inserts a R3 understandable FA129 sid, and let the packet tunnel through like TI-LFA. R4 can then take 406xxx label and apply QOS for VFA600.

R2 is the PLR.

8.3.3. Option 3: New CSPF

R1, R2, R4 and R5 detects that R3 is not participating in VFA600. They could spawn a CSPF instead of using result from FA129. However, this approach uses more processing power, and it is the least preferred option. It might create disconnection network due to misconfiguration. For example, it is only a misconfiguration in R3. R3 indeed could support the new extension. Now, the VFA600 is disconnected between R1 and R5.

8.3.4. Choice of options

In this draft, Option 1 is mandatory for implementation and is turned on by default. It could be turned off for strict compliance.

Option 2 and option 3 are optional.

Option 1 and Option 2 could co-exist during transition phase.

R2, adjacent to R3, is the PLR. R4 is the PLR for the reverse direction.

9. Error Handling

If a node detects another node which is supposed to participate in certain VFA but it is not, a warning should be displayed or logged. The invalid configuration combinations are shown in section 6.

For the node which is adjacent to the misconfigured node, it should possess the capability mentioned in Section 8 on compatibility .

10. Security Consideration

TBD

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

11.2. Informative References

[RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <https://www.rfc-editor.org/info/rfc8402>.

[RFC8665] Psenak, P., Ed., Previdi, S., Ed., Filfils, C., Gredler, H., Shakir, R., Henderickx, W., and J. Tantsura, "OSPF Extensions for Segment Routing", RFC 8665, DOI 10.17487/RFC8665, December 2019, <https://www.rfc-editor.org/info/rfc8665>.

[RFC8666] Psenak, P., Ed. and S. Previdi, Ed., "OSPFv3 Extensions for Segment Routing", RFC 8666, DOI 10.17487/RFC8666, December 2019, <https://www.rfc-editor.org/info/rfc8666>

[RFC8667] Previdi, S., Ed., Ginsberg, L., Ed., Filsfils, C., Bashandy, A., Gredler, H., and B. Decraene, "IS-IS Extensions for Segment Routing", RFC 8667, DOI 10.17487/RFC8667, December 2019, <https://www.rfc-editor.org/info/rfc8667>.

[TI-LFA] Litkowski, S., Bashandy, A., Filsfils, C., Francois, P., Decraene, B., and D. Voyer, "Topology Independent Fast

Reroute using Segment Routing",
<https://datatracker.ietf.org/doc/html/draft-ietf-rtgwg-segment-routing-ti-lfa>

[FLEXAGLO] S. Hegde, P. Psenak and etc, IGP Flexible Algorithm
https://datatracker.ietf.org/doc/draft-ietf-lsr-flex-algo

[ISO10589] International Organization for Standardization, "Information technology -- Telecommunications and information exchange between systems -- Intermediate system to Intermediate system intra-domain routeing information exchange protocol for use in conjunction with the protocol for providing the connectionless-mode network service (ISO 8473)", ISO/IEC 10589:2002, Second Edition, November 2002.

12. Acknowledgments

The following people have contributed to this document:
Shraddha Hegde, Juniper Networks

Authors' Addresses

Louis Chan (Editor)
Juniper Networks Group (Hong Kong) Limited
Suites 3001-7
30th Floor Tower 2
Times Square
1 Matheson Street
Causeway Bay
Hong Kong

Phone: +852-38562100
Email: louisc@juniper.net

Krzysztof Grzegorz Szarkowicz
Juniper Networks
Parkring 10
A-1010 Wien
Austria

Phone: +49 89 203012127
Email: kszarkowicz@juniper.net

Gyan Mishra
Verizon Inc.

Email: gyan.s.mishra@verizon.com