MPLS Working Group
Internet-Draft
Intended status: Standards Track
Expires: December 30, 2017

S. Bryant
X. Xu
M. Chen
Huawei
A. Farrel
J. Drake
Juniper Networks

June 28, 2017

A Unified Approach to IP Segment Routing
draft-bryant-mpls-unified-ip-sr-00

Abstract

Segment routing is a new and powerful forwarding paradigm that allows packets to be steered through a network on paths other than the shortest path derived from the routing protocol. The approach uses information encoded in the packet header and does not make use of a signaling protocol to pre-install paths in the network.

Two different encapsulations have been defined to enable segment routing in an MPLS network and in an IPv6 network. While acknowledging that there is a strong need to support segment routing in both environments, this document defines a converged, unified approach to segment routing that enables a single mechanism to be applied in both types of network. The resulting approach is also applicable to IPv4 networks without the need for any changes to the IPv4 specification.

This document makes no changes to the segment routing architecture and builds on existing protocol mechanisms such as the encapsulation of MPLS within UDP defined in RFC 7510.

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 December 30, 2017.

Copyright Notice

Copyright (c) 2017 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

The approach to IPv6 segment routing (SR) described in [I-D.ietf-6man-segment-routing-header] can be challenging to implement in some types of forwarder, particularly where a large number of instructions/segments are needed to specify the required behaviour. Furthermore, the approach does not allow the use of SR techniques in legacy IPv4 networks. In this document we describe a low overhead method for running SR in IP networks by using an MPLS label stack carried in UDP as a method of encoding the segment routing instructions to be executed as the packet traverses the network. We call this Unified Segment Routing (USR).

The method defined is a complementary way of running SR in an IP network when compared to [I-D.ietf-6man-segment-routing-header]. Implementers and deployers should consider the benefits and drawbacks of each method and only select the approach defined here where its properties are beneficial.

The format that we propose requires 32 bits per additional instruction compared to the 128 bits for the method described in [I-D.ietf-6man-segment-routing-header]. The methods are further compared in Section 10.

Although the segment routing instructions (i.e., the segment identifiers) are encoded as MPLS labels, this is a hardware convenience rather than an indication that the whole MPLS protocol stack and in particular the MPLS control protocols need to be deployed. It is a hardware convenience because many hardware components are already able to perform lookups based on MPLS labels.

2. Requirements Language

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

3. The Unified Segment Routing Protocol Stack

The USR protocol stack is shown below in Figure 1.

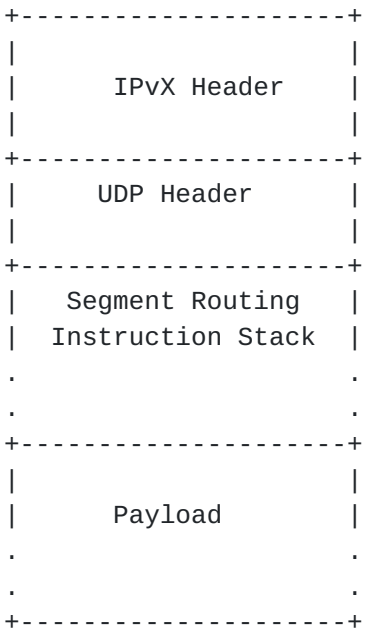


Figure 1: Packet Encapsulation

The payload may be of any type that, with an appropriate convergence layer, can be carried over a packet network. It is anticipated that the most common packet types will be IPv4, IPv6, native MPLS and pseudowires [RFC3985].

Preceding the Payload is the Segment Routing Instruction Stack (SRIS) that carries the sequence of instructions to be executed on the packet as it traverses the network. This is the Segment Identifier (SID) stack.

Preceding the SRIS is a UDP header. The UDP header is included to:

- Introduce entropy to allow equal-cost multi-path load balancing (ECMP) [RFC2992] in the IP layer [RFC7510].
- Provide a protocol multiplexing layer as an alternative to using a new IP type/next header.
- Allow transit through firewalls and other middleboxes.
- Provide disagregation.

Preceding the UDP header is the IP header which may be IPv4 or IPv6.

4. The Segment Routing Instruction Stack

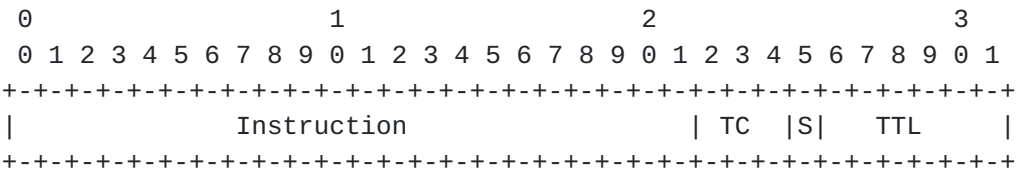
The core of the protocol encapsulation is the Segment Routing Instruction Stack (SRIS). This consists of a stack of Segment Identifiers as described in [I-D.ietf-spring-segment-routing] encoded as an MPLS label stack as described in [I-D.ietf-spring-segment-routing-mpls].

The top SRIS entry is the next instruction to be executed. When the node to which this instruction is directed has processed the instruction it is removed (popped) from the SRIS, and the next instruction processed.

Each Instruction is encoded in a single Label Stack Entry (LSE) as shown in Figure 2. The basic structure of the LSE is unchanged from [RFC3032] and the meanings of the Traffic Class, Bottom of Stack, and Time to Live fields are also unchanged.

4.1. SRIS Format

As described in [I-D.ietf-spring-segment-routing-mpls], the SRIS uses the same format as [RFC3032]. This is a compact representation that allows the use of existing data plane hardware. Its use does not imply that MPLS needs to be enabled, or that MPLS protocols need to be used. It is simply a compact, convenient way of carrying the instructions (the SIDs) needed to direct how the packet traverses the network.



Instruction: Label Value, 20 bits
TC: Traffic Class, 3 bits
S: Bottom of Stack, 1 bit
TTL: Time to Live, 8 bits

Figure 2: SRIS Label Stack Entry

As with [I-D.ietf-spring-segment-routing-mpls] a 32 bit LSE is used to carry each SR instruction. The instruction itself is carried in the 20 bit Label Value field. The TC field has the normal meaning as defined in [RFC3032] and modified in [RFC5462]. The S bit has bottom of stack semantics defined in [RFC3032]. TTL is discussed in Section 4.2.

4.2. TTL

The setting of the TTL is application specific, but the following operational consideration should be born in mind. In SR the size of the label stack may be increased within a single routing domain by various operations such as the pushing of a binding SID. Furthermore in SR packets are not necessarily constrained to travel on the shortest path with that routing domain. Consideration therefore has to be given to possibility of a forwarding loop. To mitigate against this it is RECOMMENDED that the TTL is continuously decremented as the packet passes through the SR network regardless of any other changes to the network layer encapsulation.

5. UDP/IP Encapsulation.

The procedures defined in [RFC7510] are followed. RFC7510 specifies the values to be used in the UDP Source Port, Destination Port, and Checksum fields.

An administrative domain, or set of administrative domains that are sufficiently well managed and monitored to be able to safely use IP segment routing is likely to comply with the requirements called out in [RFC7510] to permit operation with a zero checksum over IPv6. However each operator needs to validate the decision on whether or not to use a UDP checksum for themselves.

The [RFC7510] UDP header may be carried over IPv4 or over IPv6.

The IP source address is the address of the encapsulating device. The IP destination address is implied by the instruction at the top of the instruction stack.

If IPv4 is in use fragmentation is not permitted.

6. Metadata

There are a number of ways that metadata could be carried:

- The metadata could be included in the packet.
- A SID could point to locally stored metadata.
- Metadata could be carried in a support packet that does not include any user data.
- Metadata could be provided out of band.

Metadata is for future study.

7. Elements of Procedure

There are six type of node in an SR domain:

- Domain ingress nodes that receive packets and encapsulate them for transmission across the domain. These packets may be native IP packets or may already be SR packets.
- Legacy transit nodes that are IP routers but are not able to perform segment routing.
- Transit nodes that are SR capable but that are not identified by a SID in the SID stack.
- Transit nodes that are SR capable and need to perform SR routing.
- The penultimate SR capable node on the path that processes the last SID on the stack.
- The domain egress node that forwards the payload packet for ultimate delivery.

The following sub-sections describe the processing behavior in each case.

7.1. Domain Ingress

Domain ingress nodes receive packets from outside the domain and encapsulate them to be forwarded across the domain. Received packets may already be MPLS-SR packets (in the case of connecting two MPLS-SR networks across a native IP network) or may be IP or MPLS packets.

In the latter case, the packet is classified by the domain ingress node and an MPLS-SR stack is imposed. In the former case the MPLS-SR stack is already in the packet. The top entry in the stack is popped from the stack and retained for use below.

The packet is then encapsulated in UDP with the destination port set to 6635 to indicate "MPLS-UDP" as described in [RFC7510]. The source UDP port is set randomly or to provide entropy as described in [RFC7510].

The packet is then encapsulated in IP for transmission across the network. The IP source address is set to the domain ingress node, and the destination address is set to the address corresponding to the label that was previously popped from the stack.

The packet is then sent into the IP network routing the packet according to the local FIB and applying hashing to resolve any ECMP choices.

7.2. Legacy Transit

A legacy transit node is an IP router that has no SR capabilities. When such a router receives an MPLS-SR-in-UDP packet it will carry out normal TTL processing and if the packet is still live it will forward it as it would any other UDP-in-IP packet. The packet will be routed toward the destination indicated in the packet header using the local FIB and applying hashing to resolve any ECMP choices.

7.3. On-Path Pass-Through SR Nodes

Just because a node is SR capable and receives an MPLS-SR-in-UDP packet does not mean that it performs SR processing on the packet. Only routers identified by SIDs in the SR stack need to do such processing.

Routers that are not addressed by the destination address in the IP header simply treat the packet as a normal UDP-in-IP packet carrying out normal TTL processing and if the packet is still live routing the packet according to the local FIB and applying hashing to resolve any ECMP choices.

7.4. SR Transit Nodes

When a router receives an MPLS-SR-in-UDP packet that is addressed to it, it acts as follows:

- Perform TTL processing as normal for an IP packet
- Determine that the packet is addressed to the local node
- Find that the payload is UDP and that the destination port indicates MPLS-in-IP
- Strip the IP and UDP headers
- Pop the top label from the SID stack and retain it for use below
- Encapsulate the packet in UDP with the destination port set to 6635 and the source port set for entropy.
- Encapsulate the packet in IP with the IP source address set to this transit router, and the destination address set to the

address corresponding to the label that was previously popped from the stack.

- Send the packet into the IP network routing the packet according to the local FIB and applying hashing to resolve any ECMP choices.

7.5. Penultimate SR Transit

The penultimate SR transit node is only different from the SR transit node described in Section 7.4 because it pops the final MPLS-SR SID from the stack. In order to avoid confusion at the egress, the router replaces the popped SR label with an explicit null label (label value 0 [RFC3032]). The packet is then encapsulated and sent as described in Section 7.4.

7.6. Domain Egress

The domain egress strips the IP and UDP headers, pops the explicit null label, and forwards the payload packet according to its type and the local routing/forwarding mechanisms.

8. Modes of Deployment

As previously noted, the procedures described in this document may be used to connect islands of SR functionality across an IP backbone, or can provide SR function within a native IP network. This section briefly expounds upon those two deployment modes.

8.1. Interconnection of SR Domains

Figure 3 shows two SR domains interconnected by an IP network. The procedures described in this document are deployed at border routers R1 and R2 and packets are carried across the backbone network in a UDP tunnel.

R1 acts as the domain ingress as described in Section 7.1. It takes the MPLS-SR packet from the SR domain, pops the top label and uses it to identify its peer border router R2. R1 then encapsulates the packet in UDP in IP and sends it toward R2.

Routers within the IP network simply forward the packet using normal IP routing.

R2 acts as a domain egress router as described in Section 7.6. It receives a packet that is addressed to it, strips the IP and UDP headers, and acts on the payload SR label stack to continue to route the packet.

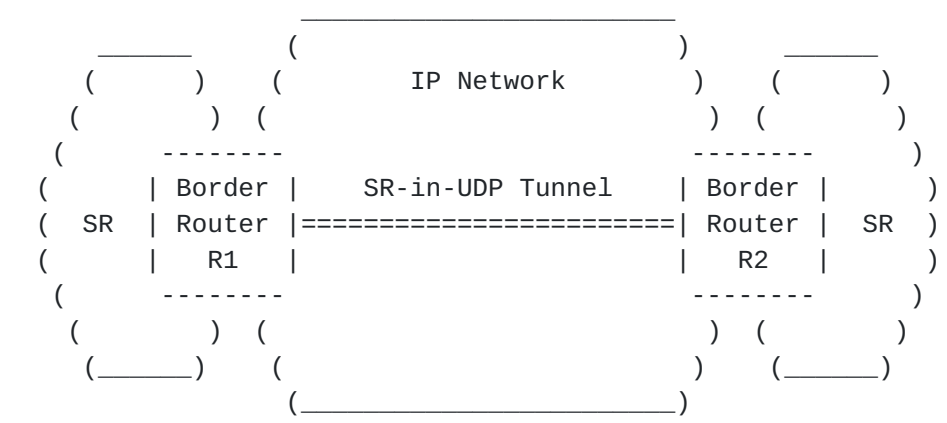


Figure 3: SR in UDP to Tunnel Between SR Sites

8.2. SR Within and IP Network

Figure 4 shows the procedures defined in this document to provide SR function across an IP network.

R1 receives a native packet and classifies determining that it should be sent on the SR path R2-R3-R4-R5. It imposes a label stack accordingly and then acts as a domain ingress as described in Section 7.1. It pops the label for R2, and encapsulates the packet in UDP in IP, sets the IP source to R1 and the IP destination to R2, and sends the packet into the IP network.

Routers Ra and Rb are transit routers that simply forward the packets using normal IP forwarding. They may be legacy transit routers (see Section 7.2) or on-path pass-through SR nodes (see Section 7.3).

R2 is an SR transit nodes as described in Section 7.4. It receives a packet addressed to it, strips the IP and UDP headers, and processes the SR label stack. It pops the top label and uses it to identify the next SR hop which is R3. R2 then encapsulates the packet in UDP in IP setting the IP source to R2 and the IP destination to R3.

Rc, Rd, and Re are transit routers and perform as Ra and Rb.

R3 is an SR transit nodes and performs as R2.

R4 is a penultimate SR transit node as described in Section 7.5. It receives a packet addressed to it, strips the IP and UDP headers, and processes the SR label stack. It pops the top label and uses it to identify the next SR hop which is R5. This was the last label in the stack so R4 includes an explicit null label before encapsulating the packet in UDP in IP setting the IP source to R4 and the IP destination to R5.

R5 is the domain egress as described in Section 7.6. It receives a packet addressed to it, strips the IP and UDP headers, and pops the explicit null label before forwarding the payload packet.

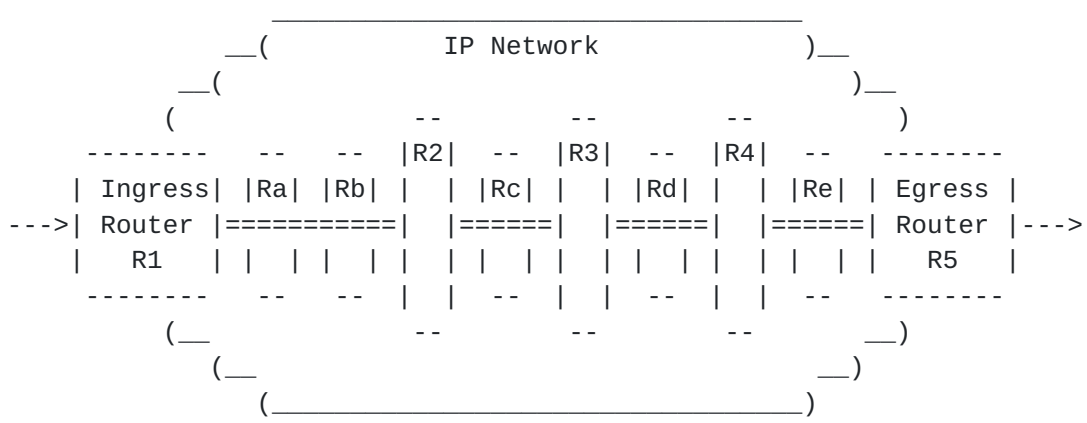


Figure 4: SR Within an IP Network

9. OAM

OAM at the payload layer follow the normal OAM procedures for the payload. To the payload the whole SR network looks like a tunnel.

OAM in the IP domain follows the normal IP procedures. This can only be carried out between IP hops.

OAM between instruction processing entities i.e. at the SR layer uses the procedures documented for MPLS.

10. Comparison with SRv6

The format described in [I-D.ietf-6man-segment-routing-header] hereon referred to as SRv6 requires an initial 36 octet IPv6 header but cannot support IPv4. USR requires either am initial 36 octet IPv6 header of an initial 20 octet IPv4 header.

o SRv6 requires an 8 octet SR header, USR requires a UDP header which is also 8 octets.

o SRv6 requires 16 octets per SID, whereas USR requires only 4 octets per SID.

o The SRv6 SIDs can be a global identifier, but the USR SIDs cannot be.

o SRv6 has an extension to support path repair at the SR level. This is for further study with USR.

o SRv6 includes the intended path history in the packet. USR would require the path history to be added as metadata.

11. Security Considerations

It is difficult for an attacker to pass a raw MPLS encoded packet into a network and operators have considerable experience at excluding such packets.

It is easy for an ingress node to detect any attempt to smuggle IP packet into the network since it would see that the UDP destination port was set to MPLS. As noted in Section 6 legitimate packets for SR processing within the network could be signed. SR packets not having a destination address terminating in the network would be transparently carried and would pose no security risk to the network under consideration.

The security consideration of [I-D.ietf-spring-ipv6-use-cases] and [RFC7510] apply.

12. IANA Considerations

This document makes no IANA requests.

13. Acknowledgements

This draft was inspired by [I-D.xu-mpls-unified-source-routing-instruction], and we acknowlegde the following authors of that draft: Robert Raszuk, Uma Chunduri, Luis M. Contreras, Luay Jalil, Hamid Assarpour, Gunter Van De Velde, Jeff Tantsura, and Shaowen Ma.

14. References

14.1. Normative References

[I-D.ietf-spring-segment-routing]
Filsfils, C., Previdi, S., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", draft-ietf-spring-segment-routing-12 (work in progress), June 2017.

[I-D.ietf-spring-segment-routing-mpls]
Filsfils, C., Previdi, S., Bashandy, A., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing with MPLS data plane", draft-ietf-spring-segment-routing-mpls-10 (work in progress), June 2017.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC3032] Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y., Farinacci, D., Li, T., and A. Conta, "MPLS Label Stack Encoding", RFC 3032, DOI 10.17487/RFC3032, January 2001, <http://www.rfc-editor.org/info/rfc3032>.

[RFC5462] Andersson, L. and R. Asati, "Multiprotocol Label Switching (MPLS) Label Stack Entry: "EXP" Field Renamed to "Traffic Class" Field", RFC 5462, DOI 10.17487/RFC5462, February 2009, <http://www.rfc-editor.org/info/rfc5462>.

[RFC7510] Xu, X., Sheth, N., Yong, L., Callon, R., and D. Black, "Encapsulating MPLS in UDP", RFC 7510, DOI 10.17487/RFC7510, April 2015, <http://www.rfc-editor.org/info/rfc7510>.

14.2. Informative References

[I-D.ietf-6man-segment-routing-header]
Previdi, S., Filsfils, C., Raza, K., Leddy, J., Field, B., daniel.voyer@bell.ca, d., daniel.bernier@bell.ca, d., Matsushima, S., Leung, I., Linkova, J., Aries, E., Kosugi, T., Vyncke, E., Lebrun, D., Steinberg, D., and R. Raszuk, "IPv6 Segment Routing Header (SRH)", draft-ietf-6man-segment-routing-header-06 (work in progress), March 2017.

[I-D.ietf-spring-ipv6-use-cases]
Brzozowski, J., Leddy, J., Filsfils, C., Maglione, R., and M. Townsley, "IPv6 SPRING Use Cases", draft-ietf-spring-ipv6-use-cases-11 (work in progress), June 2017.

[I-D.xu-mpls-spring-islands-connection-over-ip]
Xu, X., Raszuk, R., Chunduri, U., Contreras, L., and L. Jalil, "Connecting MPLS-SPRING Islands over IP Networks", draft-xu-mpls-spring-islands-connection-over-ip-00 (work in progress), October 2016.

[I-D.xu-mpls-unified-source-routing-instruction]
Xu, X., Bryant, S., Raszuk, R., Chunduri, U., Contreras, L., Jalil, L., Assarpour, H., Velde, G., Tantsura, J., and S. Ma, "Unified Source Routing Instruction using MPLS Label Stack", draft-xu-mpls-unified-source-routing-instruction-01 (work in progress), June 2017.

[RFC2992] Hopps, C., "Analysis of an Equal-Cost Multi-Path Algorithm", RFC 2992, DOI 10.17487/RFC2992, November 2000, <http://www.rfc-editor.org/info/rfc2992>.

[RFC3985] Bryant, S., Ed. and P. Pate, Ed., "Pseudo Wire Emulation Edge-to-Edge (PWE3) Architecture", RFC 3985, DOI 10.17487/RFC3985, March 2005, <http://www.rfc-editor.org/info/rfc3985>.

Authors' Addresses

Stewart Bryant
Huawei

Email: stewart.bryant@gmail.com

Xiaohu Xu
Huawei

Email: xuxiaohu@huawei.com

Mach Chen
Huawei

Email: mach.chen@huawei.com

Adrian Farrel
Juniper Networks

Email: afarrel@juniper.net

John Drake
Juniper Networks

Email: jdrake@juniper.net