SRv6: Is There Anybody Out There?

Victor-Alexandru Pădurean
MPI-INF
vpadurea@mpi-inf.mpg.de

Oliver Gasser
MPI-INF
oliver.gasser@mpi-inf.mpg.de

Randy Bush
Arrcus / IIJ
randy@psg.com

Anja Feldmann
MPI-INF
anja@mpi-inf.mpg.de

***Abstract*—Segment routing is a modern form of source-based routing, i.e., a routing technique where all or part of the routing decision is predetermined by the source or a hop on the path. Since initial standardization efforts in 2013, segment routing seems to have garnered substantial industry and operator support. Especially segment routing over IPv6 (SRv6) is advertised as having several advantages for easy deployment and flexibility in operations in networks. Many people, however, argue that the deployment of segment routing and SRv6 in particular poses a significant security threat if not done with the utmost care.**

In this paper we conduct a first empirical analysis of SRv6 deployment in the Internet. First, we analyze SRv6 behavior in an emulation environment and find that different SRv6 implementations have the potential to leak information to the outside. Second, we search for signs of SRv6 deployment in publicly available route collector data, but could not find any traces. Third, we run large-scale traceroute campaigns to investigate possible SRv6 deployments. In this first empirical study on SRv6 we were unable to find traces of SRv6 deployment even for companies that claim to have it deployed in their networks.

***Index Terms*—segment routing, IPv6, SRv6**

1. Introduction

Since its initial specification, segment routing (SR) has attracted attention from big vendors in the networking industry such as Cisco [1], Juniper [2], Huawei [3] [4], and Nokia [5]. SR is a technique which leverages the source-routing principle, allowing a sender or an intermediate node to specify (in part) the path a packet takes on its way. In addition, SR brings many promises [6], one important benefits being the fact that there is no need to keep per-application and per-flow state, as all the necessary information is stored in the packet itself. It also enables software-defined networking (SDN) and the selective use of different network appliances such as firewalls or snort [7]. Another benefit of SR mentioned by networking companies is its support of resilience techniques such as Topology Independent Loop-free Alternate Fast Re-route (TI-LFA) [8], which in a failure scenario allows quickly shifting traffic to a backup path.

Moreover, many networking vendors claim to be working on adding SR-support into their products. They also started offering support to their clients to integrate SR into their networks. Such clients include SoftBank [9], Line Corporation [10], MTN Uganda [4], Indosat Ooredoo [11], Rakuten [12], Bell Canada [13], China Unicom [14], China Telecom [4], Iliad [15] and CERNET2 [4].

Recently, segment routing also started to attract the attention of the research community, with papers describing how new functions can be implemented on top of segment routing [16] [17] [18] [19], review standardization activities, and review articles dedicated to segment routing [20].

Furthermore, since SR is a relatively new technology compared to e.g., MPLS [21], it is not yet as mature. Thus, network operators engage in lively discussion about the benefits and possible downsides such as security implications of large-scale SR deployment, especially SR over IPv6 (SRv6) [22]. Security issues may also arise from the uninformed use of SRv6 in a network [23]. Moreover, the IETF community is also looking into security issues related to SRv6 deployment [24]. Therefore, in this paper we aim to investigate the current state of SRv6 deployment in the wild.

2. Background and Related Work

Segment routing is based on the concept of source routing, i.e., it allows a source node or an intermediate node to predetermine (part of) the path taken towards a destination [25]. Segment routing, however, goes beyond a pure list of forwarding instructions: It can chain services and obtain complex behaviors as a solution for service differentiation. A *segment* is the basis for segment routing. It is composed of a *locator* (i.e., a unique identifier of the network node where the instruction should be executed), an *instruction* the node executes, either topological (i.e., forwarding) or requiring a service to be executed, and optional *arguments* for the instruction. Segments can be recognized by their SR Segment Identifiers (SIDs). We can chain segments into a list which connects the *ingress node* (i.e., headend, where the packet becomes "SR-aware") to the *egress node* (i.e., endpoint, either the final destination of the packet or the node where the SR capabilities are removed from a packet). This list of segments is called an *SR path*. The set of nodes a packet travels through between the ingress and egress nodes constitutes an *SR domain*. In Figure 1 we depict these concepts in an example topology.

Furthermore, an *SR policy* describes how traffic is handled within an SR domain [27]. A policy is identified through the tuple `<headend, color, endpoint>`. The color is used to differentiate between policies that have the same headend and endpoint. Note that a policy can have multiple SR paths available (e.g., for redundancy). In this case a preference determines which SR policy is being picked.

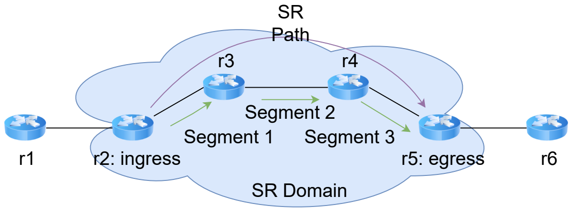


Figure 1: Segment routing topology example showing SR domain, ingress, egress, SR path, and segments [26].

In order to distribute SR policies we can make use of BGP update messages [28]. BGP speakers can propagate candidate paths for SR policies by making use of Color Extended Communities.

On the protocol level, segment routing works on top of MPLS or IPv6. Due to the long history of using MPLS in networking [29] [30] [21] [31], SR-MPLS is thought to be more mature than SR over IPv6 [32]. In this paper we focus on the prevalence of SR implemented on top of IPv6, i.e., *SRv6*. This is done through a new type of routing header (next header value 43) called *Segment Routing Header*, with routing type 4 [33]. The details regarding this header can be seen in Figure 2. The inner workings of SRv6 are described in the corresponding RFC [34].

Lately, researchers started exploring the capabilities and implications of segment routing.

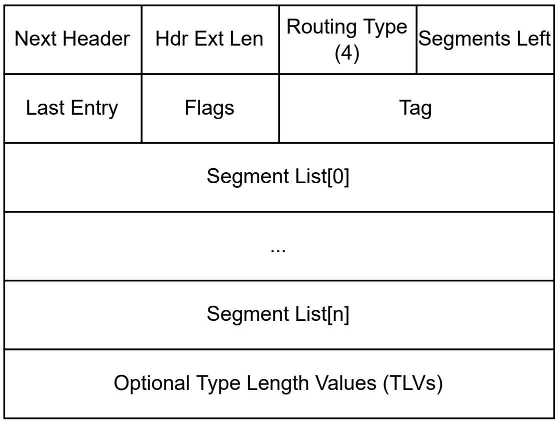
Ventre et al. conduct a comprehensive survey on the introduction, motivation, and evolution of SR, SR-MPLS, and SRv6 [20]. The authors provide a classification of current SR activities SR. They conclude that SR's potential is not yet fully leveraged. For example, they find a lack of works focusing on failure monitoring, but they seem confident that the network programming capabilities of SR will get more attention in the coming years.

Tulumello et al. propose ideas on how to make SRv6 more efficient by reducing the size of SIDs [16]. Instead of representing only one segment with a single SID, they encode up to six micro instructions, represented through micro SIDs. This drastically reduces the SIDs needed to be present in the SRH, thus reducing the packet overhead of SRv6 containing many SIDs.

Mayer et al. present ideas on how SRv6 can be integrated in an IoT and cloud infrastructure leading to a new distributed processing model [17]. They theorize an abstract computing machine, which they call an SR-IoT Computing Machine which can be programmed with its own instruction set. Thus, by building their own toolchain and leveraging the capabilities of segment routing, the authors treat the whole Cloud-IoT network (i.e., without differentiation between core and edge) as a single logical machine. This way a developer focuses on the application logic, while the owner of the infrastructure enforces management policies.

Lebrun and Bonaventure describe their experience implementing SRv6 in the Linux kernel [35]. In addition, they analyze the performance impact of SRH insertion. They find that the performance overhead for SR entry points with their implementation for Linux 4.10 is limited to 15%.

To the best of our knowledge this work is the first conducting an empirical analysis of the SRv6 landscape in the wild.



Next Header	Hdr Ext Len	Routing Type (4)	Segments Left
Last Entry	Flags	Tag	
Segment List[0]			
...			
Segment List[n]			
Optional Type Length Values (TLVs)			

Figure 2: SRv6 header structure.

3. Emulation

Before analyzing the deployment of SRv6 in BGP and with traceroute measurements, we set up a controlled emulation environment in a lab. We use IPMininet [36], a Python library that extends Mininet [37], enabling the use of SRv6 and BGP. IPMininet directly uses the Linux kernel's implementation of segment routing. The setup is simple, yet complex enough to emphasize the traces that SRv6 may leave in an multi-AS environment. We worked in an IPv6-only setting. Figure 3 shows our emulation topology. We use one main AS—namely AS1—with 4 routers (i.e., r2, r3, r4, and r5), forming an iBGP full mesh. AS1 also contains one host system: h2, connected to r5. Moreover, we set up a smaller AS—namely AS2—containing only one router (i.e., r1). There is also a host in AS2 (i.e., h1) which is connected to the router r1. The edge routers r1 and r2 are exchanging reachability information through an eBGP session. Note that all routers run both OSPF and BGP. Finally, we enable SRv6 on all the nodes and set up r2 as an SR entry point which enforces a simple SR policy: when receiving a packet destined to h2, r2 should forward it through r4.

This policy can be implemented by choosing one of the two SRv6 modes: `H.Insert` (called `inline` in the Linux Kernel implementation) or `H.Encap` (`encap` in Linux). The `encap` mode indicates that the incoming packet will be encapsulated into another IPv6 packet (i.e., IPv6-in-IPv6) at the ingress node (i.e., r1). The source of the outer packet is set to r2 and the destination address is set to the next segment (i.e., r4 in our case, which is also our egress node). Moreover, the entry point will add a segment routing header (SRH). Note that in our example here, the segment r4 in the SRH can be omitted if a reduced SRH is being used [33]. The egress node will strip the packet of the outer IPv6 layer and forward it to the original destination (i.e., h2).

The `inline` mode, however, aims to modify the IPv6 layer of the original packet. It changes the destination address, adds an SRH, but leaves the source address untouched. This means that an ICMPv6 message (e.g., due to Hop Limit exceeded) will reach the original source [35], allowing detection of SRv6 deployment with traceroute.

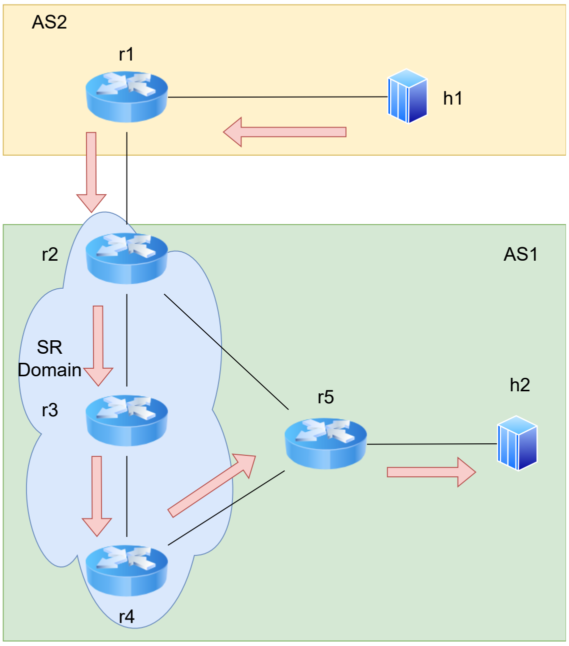


Figure 3: SRv6 emulation topology showing two ASes, one of which contains an SR domain with an SR path. The arrows show a packet's path.

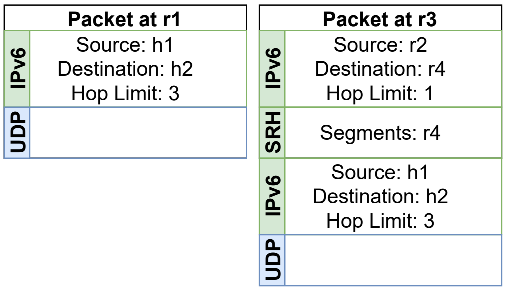
Our findings confirm this. We send `traceroute` probes from h1 to h2, capturing all the traffic at h1. When using `encap`, no packet contained traces of SRv6, but when using `inline`, we can see SRHs in the ICMPv6 packets returned from the SR domain. This is a promising finding, as various parties claim to have implemented and use `H.Insert` and `H.Insert.Red` (i.e., reduced insertion) [38].

To summarize: Our emulation experiments show that it is possible to detect SRv6 deployment with traceroute, if SRv6 is implemented with inline mode.

4. Investigating BGP Route Collectors

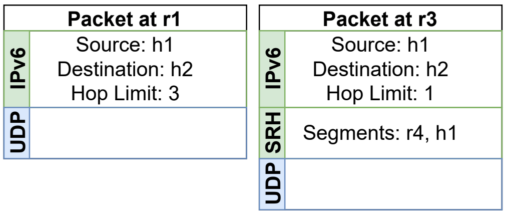
One possible source of SRv6 leakage is BGP. We inspect BGP data from 10 popular route collectors from RIPE RIS and RouteViews (see Table 1 for a list of RCs) in September 2021.

We find that Color Extended BGP Communities [39] can be used to steer traffic according to various SR policies [28], [40]. Thus we explore data from BGP archives looking for these types of communities. This can be done by detecting the specific bytes in the Type and Sub-Type fields of extended communities: `0x03` and `0x0b`. Our search for these byte combinations does not yield results in the BGP collector data.


Packet at r1
IPv6: Source: h1, Destination: h2, Hop Limit: 3
UDP

Packet at r3
IPv6: Source: r2, Destination: r4, Hop Limit: 1
SRH: Segments: r4
IPv6: Source: h1, Destination: h2, Hop Limit: 3
UDP

Figure 4: Packet structure and content for `H.Encap` behavior at routers r1 and r3.


Packet at r1
IPv6: Source: h1, Destination: h2, Hop Limit: 3
UDP

Packet at r3
IPv6: Source: h1, Destination: h2, Hop Limit: 1
SRH: Segments: r4, h1
UDP

Figure 5: Packet structure and content for `H.Insert` behavior at routers r1 and r3.

Next, we also search for other communities that may be an indicator of SRv6 usage, directly in the BGP data. Then, we collect a list of ASes that belong to organizations that claim to be using SR (we will call them SR-suspect ASes). We then follow a simple procedure for inspecting whether these SR-suspect ASes have specific communities in common. First, we extract all BGP communities that appear with at least one SR-suspect AS on the AS path. Subsequently, we extract the communities that never appear with SR-suspect ASes on the path. Finally, we make a difference of both of these sets of BGP communities and obtain a list of communities which only appear in BGP announcements with SR-suspect ASes on the path. We then count in how many SR-suspect ASes each community appears, aiming to find a community used by as many SR-suspect ASes as possible. However, each community we find is only used by a small number of ASes. As can be seen in Table 1, the majority of communities is used by only a single AS, whereas a small number is used by two or more ASes. We further inspect the cases where BGP communities are seen in announcements with more than one SR-suspect AS. We find that these cases are almost all related to sibling ASes, i.e., two ASes with a different ASN but operated by the same organization. Only at `rv4` we find three communities within a single BGP announcement with two suspect ASes on the AS path. We find that those communities are related to tagging [41] and peer selection [42], and not SRv6.

Moreover, we investigate whether the communities found only for SR-suspect ASes are location communities [43]. We find that between 25% to 50% of all communities from Table 1 are identified by the BGP location community database [44] as location communities. This shows that a non-negligible portion of these communities is very unlikely to be used for SR.

Furthermore, we inspect BGP path attributes as potential indicators of SR deployment. These include `BGP Prefix-SID`, `BGP-LS Attribute` and `Tunnel Encapsulation`. The `BGP Prefix-SID` attribute [45] is a BGP extension used for signaling information about BGP Prefix Segment Identifiers, thus it is a good indicator of SR deployment. `BGP-LS` [46], on the other hand, is used for more general purposes, yet it has been extended [47] to support carrying SR information between ASes via BGP. This also applies to the `Tunnel Encapsulation` attribute [48], which permits carrying Prefix-SID-related information in one of its Sub-TLVs. We search the BGP collector archives

TABLE 1: Unique BGP community values seen only at suspect ASes (# comms), seen only at multiple suspect ASes (# multiple comms), and seen only at multiple suspect ASes while ignore sibling communities (# non-sibling comms).

Collector	# comms	# multiple comms	# non-sibling comms
rv2	208	14	0
rv4	257	23	3
rv5	0	0	0
rv6	97	9	0
rv-amsix	190	8	0
rrc00	257	16	0
rrc01	206	11	0
rrc04	151	5	0
rrc05	171	8	0
rrc06	89	1	0

for interesting path attributes, but we only find generic attributes (e.g., `AGGREGATOR`, `ATOMIC_AGGREGATE`, `AS_PATHLIMIT`) and no SR-specific attributes.

To summarize: We analyze data collected by 10 BGP collectors, looking for attributes that could indicate SRv6 deployment. We find no such attributes. Furthermore, we conduct a thorough investigation of BGP Communities, especially Color Extended Communities, aiming to find a pattern that may indicate SRv6 usage. Again, we do not find any relevant signal. This means that these attributes are either properly filtered by BGP speakers on the path [49] or the egress routers of SRv6 deployments.

5. Tracerouting for SRv6

In order to reveal SRv6 deployment in the Internet, we ran several traceroute measurements. Further, we want to understand if we can identify SRv6 in networks which claim to have deployed it [38]. Companies claiming to deploy `H.Insert` and its variations are especially valuable to us, since those might leak information in traceroute (cf. Section 3). Those companies are Iliad, SoftBank, Line Corporation, China Unicom, China Telecom, and CERNET2. We use PeeringDB [50] and BGPView [51] to collect AS numbers related to these organizations. We then use the WHOIS database and CAIDA's Routeviews BGP data [52] to get IPv6 prefixes for the identified ASes. Finally, we generate 100 random addresses for each prefix resulting in 213.8 k addresses in total.

Next, we use Yarrp [53] to run traceroute towards each of these addresses. Before conducting active measurements we incorporate proposals by Partridge and Allman [54] and Kenneally and Dittrich [55] and follow best measurement practices [56] by using dedicated servers, informative rDNS names, a website with information, and maintaining a blocklist. We did not receive any complaints while conducting the measurements. We run the measurements from a single vantage point at MPI-INF, which might influence the traceroute paths we cover. We send two types of probes: SR-unaware probes and SR-enabled probes. The former are simply regular traceroute probes, the latter sends probes containing a segment routing header. We modify Yarrp in order to send SR-enabled probes. Sending SR-enabled probes allows us to check if SRv6-enabled routers modify the list of segments in the SRH, which would be visible in the returned quoted ICMPv6 packet. We conduct both types of measurements for various transport protocols, as shown in Table 2. We target three types of addresses: (1) We take a random sample of 10 million addresses from the IPv6 Hitlist [57] (marked as *hitlist* in Table 2). (2) We generate random addresses within prefixes of SR-suspect ASes (marked as *suspect* in tab:traceroutes). (3) We generate random addresses for each BGP-announced prefix. Even though we see SRHs being returned from the SR-enabled measurements, those are not modified at all and therefore do not leak any SRv6 deployment. Moreover, our other measurements do not return any SRH and therefore do not leak SRv6 deployment in the wild.

TABLE 2: Overview of traceroute measurements.

Date	Type	SRH sent	Targets	SRv6 leaked
2021-11-09	TCP SYN	✗	hitlist	✗
2022-02-08	TCP6 SYN	✗	suspect	✗
2022-02-09	TCP6 SYN	✓	suspect	✗
2022-02-10	ICMP6	✗	prefix	✗
2022-02-10	TCP6 ACK	✗	prefix	✗
2022-02-10	TCP6 SYN	✗	prefix	✗
2022-02-10	UDP6	✗	prefix	✗
2022-02-15	ICMP6	✗	prefix	✗
2022-02-15	TCP6 ACK	✗	prefix	✗
2022-02-17	TCP6 ACK	✗	prefix	✗

To summarize: We conduct traceroute measurements to identify SRv6 deployments in the wild, but do not see any traces of SRv6. Either companies do not use SRv6's inline mode as they claim or they filter SRHs from the returned ICMPv6 packets.

6. Conclusion

Segment routing is a protocol which has its roots in the source routing space. It brings many promises, including SDN-readiness, fast rerouting, and statelessness. Given its relatively young age, experts are concerned to the security implications it brings. Therefore, in this paper we tried to measure the SRv6 deployment leakage of segment routing in the wild. We searched for SRv6 in the real world, by exploring BGP collector archives and actively probing addresses of organizations that claim to be using SRv6. While analyzing the BGP data, we looked for path attributes that may be indicators of segment routing and tried to find a correlation between communities and segment routing usage. Unfortunately, we found no trace of segment routing. BGP speakers may be filtering such attributes, or egress routers themselves may carefully pick what kind and which part of announcements are propagated to neighbors. Finally, we conducted a traceroute measurement campaign in order to identify leaked SRv6 deployments in the Internet, but we could not find any trace. Organizations either do not use the inline mode as they claim or they carefully filter segment routing traces from the returned packets. Another possibility is that operators might use SRv6 exclusively for 5G deployments, which are likely to be more firewalled than other types of networks, possibly resulting in fewer SRv6 leaks. We release data *TODOCITE* and analysis code *TODOCITE* to be used by fellow researchers looking into investigating SRv6 deployment in the future.

Future work: We will continue to leverage BGP collector data and traceroutes in the future to see if SRv6 leaks occur. Moreover, we plan to extend the BGP analysis to cover more route collectors and longer timespans. Additionally, we will increase the target set of our traceroute analysis to discover well hidden SRv6 deployments. Finally, we plan to set up a hardware testing lab in order to extend our emulation to bare-metal devices.

References

[1] Cisco, "Segment Routing Configuration Guide, Cisco IOS XE Release 3S," https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/seg_routing/configuration/xe-3s/segrt-xe-3s-book/intro-seg-routing.html.

[2] Julian Lucek and Krzysztof Szarkowicz, "DAY ONE: CONFIGURING SEGMENT ROUTING WITH JUNOS," https://www.juniper.net/documentation/en_US/day-one-books/DO_SegmentRouting.pdf.

[3] Huawei, "CloudEngine 12800 and 12800E V200R005C10 Command Reference," https://support.huawei.com/enterprise/en/doc/EDOC1100075369/ceb6d155/segment-routing-configuration-commands.

[4] Clarence Filsfils and Kris Michielsen and Pablo Camarillo and François Clad, "SRv6," https://www.segment-routing.net/images/SRv6-TOI-rev3i-EXTERNAL.pdf, 2021.

[5] Nokia, "Segment routing in SR OS," https://www.al-enterprise.com/-/media/assets/internet/documents/ale-nokia-segment-routing-in-sr-os-ip-mpls-datasheet-en.pdf.

[6] Cisco, "About Segment Routing," https://www.cisco.com/c/en/us/td/docs/iosxr/ncs5500/segment-routing/61x/b-segment-routing-cg-ncs5500/b-segment-routing-cg-ncs5500_chapter_01.pdf.

[7] A. M. Abdelsalam, "Demo: Chaining of segment routing aware and unaware service functions," in *Networking*, 2018.

[8] S. Litkowski, A. Bashandy, C. Filsfils, P. Francois, B. Decraene, and D. Voyer, "Topology Independent Fast Reroute using Segment Routing," Internet Engineering Task Force, Internet-Draft draft-ietf-rtgwg-segment-routing-ti-lfa-08, Jan. 2022, work in Progress. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-ietf-rtgwg-segment-routing-ti-lfa-08

[9] Cisco, "Cisco Supports SoftBank on First Segment Routing IPv6 Deployment in Prep for 5G," https://newsroom.cisco.com/c/r/newsroom/en/us/a/y2019/m02/cisco-supports-softbank-on-first-segment-routing-ipv6-deployment-in-prep-for-5g.html, Feb. 2019.

[10] Hirofumi Ichihara, "LINE Data Center Networking with SRv6," https://www.segment-routing.net/conferences/2019-09-20-SRv6-LINE-DC/, Sep. 2019.

[11] Indosat Ooredoo Hutchison, "Indosat Ooredoo and Cisco to Bring SRv6 and Converged SDN Transport Network to Indonesia," https://indosatooredoo.com/portal/en/corppressreleasedetail?_id=10004335, Nov. 2020.

[12] Cisco, "Rakuten Mobile Advances Its Network for 5G and IoT Services with Cisco," https://newsroom.cisco.com/c/r/newsroom/en/us/a/y2021/m06/rakuten-mobile-advances-its-network-for-5g-and-iot-services-with-cisco.html, Jun. 2021.

[13] Yvette Kanouff, "Bell Canada implements Cisco Segment Routing," https://blogs.cisco.com/news/bell-canada-implements-cisco-segment-routing, Dec. 2017.

[14] Cisco, "China Unicom Teams with Cisco to Enable Cloud + Network Synergy with Segment Routing," https://www.globenewswire.com/news-release/2018/04/17/1479878/0/en/China-Unicom-Teams-with-Cisco-to-Enable-Cloud-Network-Synergy-with-Segment-Routing.html, Apr. 2018.

[15] ——, "Iliad Launches 5G Ready IP Network Architecture with Segment Routing IPv6 in Italy," https://newsroom.cisco.com/c/r/newsroom/en/us/a/y2019/m04/iliad-launches-5g-ready-ip-network-architecture-with-segment-routing-ipv6-in-italy.html, Apr. 2019.

[16] A. Tulumello, A. Mayer, M. Bonola, P. Lungaroni, C. Scarpitta, S. Salsano, A. Abdelsalam, P. Camarillo, D. Dukes, F. Clad *et al.*, "Micro SIDs: a solution for Efficient Representation of Segment IDs in SRv6 Networks," in *2020 16th International Conference on Network and Service Management (CNSM)*. IEEE, 2020, pp. 1–10.

[17] A. Mayer, E. Altomare, S. Salsano, F. L. Presti, and C. Filsfils, "The Network as a Computer with IPv6 Segment Routing: a Novel Distributed Processing Model for the Internet of Things," in *Proc. 1st Int. Workshop Next Gener. Oper. Syst. Cyber Phys. Syst.(NGOSCPS) CPS-IoT Week*, 2019, pp. 1–4.

[18] F. Paolucci, V. Uceda, A. Sgambelluri, F. Cugini, O. G. De Dios, V. Lopez, L. Contreras, P. Monti, P. Iovanna, F. Ubaldi *et al.*, "Interoperable multi-domain delay-aware provisioning using segment routing monitoring and bgp-ls advertisement," in *ECOC 2016; 42nd European Conference on Optical Communication*. VDE, 2016, pp. 1–3.

[19] R. Bhatia, F. Hao, M. Kodialam, and T. Lakshman, "Optimized network traffic engineering using segment routing," in *2015 IEEE Conference on Computer Communications (INFOCOM)*. IEEE, 2015, pp. 657–665.

[20] P. L. Ventre, S. Salsano, M. Polverini, A. Cianfrani, A. Abdelsalam, C. Filsfils, P. Camarillo, and F. Clad, "Segment Routing: A Comprehensive Survey of Research Activities, Standardization Efforts, and Implementation Results," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 1, pp. 182–221, 2020.

[21] E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol Label Switching Architecture," RFC 3031 (Proposed Standard), RFC Editor, Fremont, CA, USA, Jan. 2001, updated by RFCs 6178, 6790. [Online]. Available: https://www.rfc-editor.org/rfc/rfc3031.txt

[22] Aaron Gould `morgner@uni-muenster.de`, "NANOG Mailing List: SRv6," https://mailman.nanog.org/pipermail/nanog/2020-September/209650.html, Sep. 2020.

[23] Sander Steffann, "IETF Topicbox: Question from SPRING regarding draft-filsfilscheng-spring-srv6-srh-compression," https://ietf.topicbox-scratch.com/groups/ipv6/T6b0d8986abb2c65f-M74208dd2565ead3657fba489/question-from-spring-regarding-draft-filsfilscheng-spring-srv6-srh-compression, Oct. 2021.

[24] C. Li, Z. Li, C. Xie, H. Tian, and J. Mao, "Security Considerations for SRv6 Networks," Internet Engineering Task Force, Internet-Draft draft-li-spring-srv6-security-consideration-07, Oct. 2021, work in Progress. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-li-spring-srv6-security-consideration-07

[25] C. Filsfils (Ed.), S. Previdi (Ed.), L. Ginsberg, B. Decraene, S. Litkowski, and R. Shakir, "Segment Routing Architecture," RFC 8402 (Proposed Standard), RFC Editor, Fremont, CA, USA, Jul. 2018. [Online]. Available: https://www.rfc-editor.org/rfc/rfc8402.txt

[26] Juniper, "What is segment routing?" https://www.juniper.net/us/en/research-topics/what-is-segment-routing.html.

[27] C. Filsfils, K. Talaulikar (Ed.), D. Voyer, A. Bogdanov, and P. Mattes, "Draft: Segment Routing Policy Architecture," https://datatracker.ietf.org/doc/html/draft-ietf-spring-segment-routing-policy-21, Mar. 2022.

[28] S. Previdi, C. Filsfils, K. Talaulikar (Ed.), P. Mattes, D. Jain, and S. Lin, "Draft: Advertising Segment Routing Policies in BGP," https://datatracker.ietf.org/doc/html/draft-ietf-idr-segment-routing-te-policy-16, Mar. 2022.

[29] F. L. Faucheur, D. J. Heinanen, P. J. Vaananen, D. B. S. Davie, P. Cheval, S. Davari, R. Krishnan, and L. Wu, "Multi-Protocol Label Switching (MPLS) Support of Differentiated Services," RFC 3270, May 2002. [Online]. Available: https://www.rfc-editor.org/info/rfc3270

[30] J. McManus, J. Malcolm, M. D. O'Dell, D. O. Awduche, and J. Agogbua, "Requirements for Traffic Engineering Over MPLS," RFC 2702, Sep. 1999. [Online]. Available: https://www.rfc-editor.org/info/rfc2702

[31] D. Awduche, "Mpls and traffic engineering in ip networks," *IEEE Communications Magazine*, vol. 37, no. 12, pp. 42–47, 1999.

[32] Dave Bell, “NANOG Mailing List: Devil’s Advocate - Segment Routing, Why?” https://mailman.nanog.org/pipermail/nanog/2020-June/108353.html, Jun. 2020.

[33] C. Filsfils (Ed.), D. Dukes (Ed.), S. Previdi, J. Leddy, S. Matsushima, and D. Voyer, “IPv6 Segment Routing Header (SRH),” RFC 8754 (Proposed Standard), RFC Editor, Fremont, CA, USA, Mar. 2020. [Online]. Available: https://www.rfc-editor.org/rfc/rfc8754.txt

[34] C. Filsfils, P. Camarillo, J. Leddy, D. Voyer, S. Matsushima, and Z. Li, “Segment Routing over IPv6 (SRv6) Network Programming,” RFC 8986, Feb. 2021. [Online]. Available: https://www.rfc-editor.org/info/rfc8986

[35] D. Lebrun and O. Bonaventure, “Implementing IPv6 Segment Routing in the Linux Kernel,” in *Proceedings of the Applied Networking Research Workshop*, 2017, pp. 35–41.

[36] IPMininet Contributors, “IPMininet on GitHub,” https://github.com/cnp3/ipmininet, Mar. 2022.

[37] Mininet Contributors, “Mininet: An Instant Virtual Network on your Laptop (or other PC),” http://mininet.org/, Mar. 2022.

[38] S. Matsushima, C. Filsfils, Z. Ali, K. Rajaraman, and A. Dhamija, “Draft: Advertising Segment Routing Policies in BGP,” https://datatracker.ietf.org/doc/html/draft-matsushima-spring-srv6-deployment-status-13, Mar. 2022.

[39] P. Mohapatra and E. Rosen, “The BGP Encapsulation Subsequent Address Family Identifier (SAFI) and the BGP Tunnel Encapsulation Attribute,” RFC 5512 (Proposed Standard), RFC Editor, Fremont, CA, USA, Apr. 2009, obsoleted by RFC 9012. [Online]. Available: https://www.rfc-editor.org/rfc/rfc5512.txt

[40] L. Hoxha, “Traffic Engineering with Segment Routing,” https://www.ciscolive.com/c/dam/r/ciscolive/apjc/docs/2020/pdf/BRKSPG-2021.pdf, Jun. 2020.

[41] N. Hirai, “S8 service provider: Introduction to backbone design,” https://www.nic.ad.jp/ja/materials/iw/2019/proceedings/s08/s8-hirai.pdf, 2019.

[42] NTT, “Routing Policies,” https://www.gin.ntt.net/support-center/policies-procedures/routing/.

[43] B. A. Silva Jr, P. Mol, O. Fonseca, I. Cunha, R. A. Ferreira, and E. Katz-Bassett, “Automatic Inference of BGP Location Communities,” *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 6, no. 1, pp. 1–23, 2022.

[44] ——, “BGP Location Communities on GitHub,” https://github.com/TopoMapping/bgp-communities, Jan. 2022.

[45] S. Previdi, C. Filsfils, A. Lindem, A. Sreekantiah, and H. Gredler, “Segment Routing Prefix Segment Identifier Extensions for BGP,” RFC 8669, Dec. 2019. [Online]. Available: https://www.rfc-editor.org/info/rfc8669

[46] H. Gredler, J. Medved, S. Previdi, A. Farrel, and S. Ray, “North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP,” RFC 7752, Mar. 2016. [Online]. Available: https://www.rfc-editor.org/info/rfc7752

[47] S. Previdi, K. Talaulikar, C. Filsfils, H. Gredler, and M. Chen, “Border Gateway Protocol - Link State (BGP-LS) Extensions for Segment Routing,” RFC 9085, Aug. 2021. [Online]. Available: https://www.rfc-editor.org/info/rfc9085

[48] K. Patel, G. V. de Velde, S. R. Sangli, and J. Scudder, “The BGP Tunnel Encapsulation Attribute,” RFC 9012, Apr. 2021. [Online]. Available: https://www.rfc-editor.org/info/rfc9012

[49] T. Krenc, R. Beverly, and G. Smaragdakis, “AS-level BGP community usage classification,” in *Proceedings of the 21st ACM Internet Measurement Conference*, 2021, pp. 577–592.

[50] PeeringDB Contributors, “PeeringDB,” https://www.peeringdb.com/, Mar. 2022.

[51] SecurityTrails, “BGPView,” https://bgpview.io/, Mar. 2022.

[52] CAIDA, “Routeviews Prefix to AS mappings Dataset (pfx2as) for IPv4 and IPv6,” https://www.caida.org/catalog/datasets/routeviews-prefix2as/, Mar. 2022.

[53] R. Beverly, “Yarrp’ing the Internet: Randomized High-Speed Active Topology Discovery,” in *Proceedings of the 2016 Internet Measurement Conference*, 2016, pp. 413–420.

[54] C. Partridge and M. Allman, “Ethical considerations in network measurement papers,” *Communications of the ACM*, vol. 59, no. 10, pp. 58–64, 2016.

[55] E. Kenneally and D. Dittrich, “The Menlo Report: Ethical Principles Guiding Information and Communication Technology Research,” *Available at SSRN 2445102*, 2012.

[56] Z. Durumeric, E. Wustrow, and J. A. Halderman, “ZMap: Fast Internet-wide Scanning and Its Security Applications,” in *22nd USENIX Security Symposium (USENIX Security 13)*, 2013, pp. 605–620.

[57] O. Gasser, Q. Scheitle, P. Foremski, Q. Lone, M. Korczynski, S. D. Strowes, L. Hendriks, and G. Carle, “Clusters in the Expanse: Understanding and Unbiasing IPv6 Hitlists,” in *Proceedings of the 2018 Internet Measurement Conference*. New York, NY, USA: ACM, 2018.