

rtgwg
Internet-Draft
Intended status: Standards Track
Expires: 18 March 2026

D. Lamparter
NetDEF, Inc.
A. Smirnov
Cisco Systems, Inc.
J. Linkova
Google
S. Yang, Ed.
Shenzhen University
M. Xu
Tsinghua University
14 September 2025

Destination/Source Routing
draft-ietf-rtgwg-dst-src-routing-revive-04

Abstract

This note specifies using packets' source addresses in route lookups as additional qualifier to be used in hop-by-hop routing decisions. This applies to IPv6 [RFC8200] in general with specific considerations for routing protocol left for separate documents. There is nothing precluding similar operation in IPv4, but this is not in scope of this document.

Note that destination/source routing, source/destination routing, SADR, source-specific routing, source-sensitive routing, S/D routing and D/S routing are all used synonymously.

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 <https://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 18 March 2026.

Copyright Notice

Copyright (c) 2025 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 (<https://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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	3
1.1. Requirements Language	4
2. Use cases	4
2.1. Multihomed networks with provider assigned prefixes	4
2.2. Degree of traffic engineering	5
2.3. Distributed filtering based on source address	5
2.4. Walled-garden Enterprise services	6
2.5. Information Source for Neighbor Management	6
3. Principle of operation	6
3.1. Frame of reference	6
3.2. Route information and equality	7
3.3. Lookup ordering and disambiguation	7
3.4. Ordering Rationale	8
4. Routing protocol considerations	8
4.1. Source information	9
4.2. Loop-freeness considerations	9
4.3. Recursive routing	10
5. Applicability To Specific Situations	10
5.1. Recursive Route Lookups	11
5.1.1. Recursive route expansion	12
5.2. Unicast Reverse Path Filtering	12
5.3. Multicast Reverse Path Forwarding	13
5.4. Testing for Connectivity Availability	13
5.5. Performing Source Address Selection	13
5.6. Improving Source Address Selection	14
6. Interoperability	15
6.1. Interoperability in Distance-Vector Protocols	16
6.2. Interoperability in Link-State Protocols	17
7. IANA Considerations	17
8. Security Considerations	17
9. Privacy Considerations	18
10. Acknowledgements	18

11. Change Log	18
12. References	19
12.1. Normative References	19
12.2. Informative References	20
Appendix A. Implementation Options	21
A.1. Pre-expanded 2-step lookup without backtracking	21
A.2. Translation to Multi-FIB (Policy Routing) perspective	22
Appendix B. Implementation Status	24
Authors' Addresses	25

1. Introduction

Both IPv4 [RFC0791] and IPv6 [RFC8200] architectures specify that determination of the outgoing next-hop for packet forwarding is based solely on the destination address contained in the packet header. There exists class of network design problems which require packet forwarding to consider more than just the destination IP address (see Section 2 for examples).

At present these problems are routinely resolved by configuring special forwarding based on a local policy on routers. The policy enforces packet forwarding decision outcome based not only on the destination address but also on other fields in the packet's IP header, most notably the source address. Such policy-based routing is conceptually similar to static routes in that it is highly static in nature and must be closely governed via the management plane (most frequently - via managing configuration by an operator). Thus policy-based routing configuration and maintenance is costly and error-prone.

Rapid expansion of IPv6 to networks where static configuration is not acceptable due to both its static nature and necessity of frequent intervention by a skilled operator requires change in the paradigm of forwarding IP packets based only on their destination address.

This document describes architecture of destination-source routing. It includes both forwarding plane and control plane considerations and requirements. Specific considerations for particular dynamic routing protocols are outside of the scope of this note and will be covered in separate documents, for example handling of a noncontiguous sub-topology in a link-state protocol.

General concepts covered by this document are equally applicable to both IPv4 and IPv6. Considering the implementation complexity of backward compatibility of destination-source routing with traditional destination-only routing, IPv4 is left outside the scope of this document.

1.1. 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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Use cases

2.1. Multihomed networks with provider assigned prefixes

There are good reasons for networks to be multihomed - benefits of doing this may include redundancy, better performance or faster access to important resources (for example, video or cloud services) local to ISPs.

However, in a range from smaller home networks to even larger enterprises, it is likely that each service provider will assign some address space (from their PA allocation) to the network.

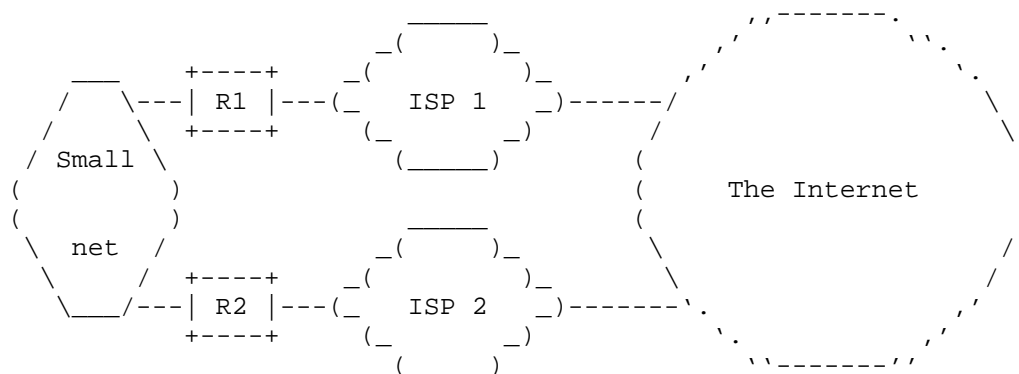


Figure 1: Example of multihomed small network

In this situation, providers are expected to perform ingress filtering according to BCP 38 [RFC2827]. This means only packets with a source address from the prefix that the provider assigned will be accepted. In addition, the assigned prefix can usually not be expected to remain the same.

In the legacy IPv4 world, NAT or policy routing would be used to produce correct behaviour. These are not desirable solutions: all variants of NAT, even the reasonably constrained NPT, break end-to-end connectivity (and may restrict concurrent use of parallel paths.) Policy routing requires a sufficiently skilled operator to manually manage these policies.

By assigning addresses from multiple prefixes each to end host (as a policy routing solution could do), the choice of uplink is left to host, including the option to choose multiple at once. Destination-source routing provides the necessary behaviour for routers (e.g. R1 and R2 in above example) to forward packets to the appropriate exit. It does so without requiring the manual configuration maintenance that policy routing would entail.

For a general introduction and aspects of interfacing routers to hosts, refer to [RFC8043].

2.2. Degree of traffic engineering

Consider enterprise consisting of a headquarter (HQ) and branch offices. A branch office is connected to the enterprise HQ network via 2 links. For performance or security reasons it is desired to route corporate traffic via one link and Internet traffic via another link. In direction branch -> HQ the problem is easily solvable by having the default route pointing to the Internet link and HQ routes pointing to another link. But destination routing does not provide an easy way to achieve traffic separation in direction HQ -> branch because destination is the same (branch network).

Source-destination routing provides an easy way to sort traffic going to the branch based on its source address.

2.3. Distributed filtering based on source address

A network has untrusted zone and secure one (and both zones comprise many links and routers). Computers from the secure zone need to be able to communicate with some selected hosts in the untrusted zone. The secure zone is protected by a firewall. The firewall is configured to check that packets arriving from the untrusted zone have destination address in the range of secure zone and source address of trusted hosts in the untrusted zone. This works but leaves the firewall open to DDOS attack from outside.

If routers in the untrusted zone are configured with destination-source routing (and, possibly, unicast RPF check) and receive via dynamic routing protocol routes <destination: secure zone; source: trusted host in the untrusted zone> then DDOS attack is dropped by

routers on the edge of destination-source routing area. DDOS attack does not even reach the firewall whose resources are freed to deal with Deep Packet Inspection. On the other hand, security policy is managed in a single point - on a router injecting relevant destination-source routes into the dynamic routing protocol.

2.4. Walled-garden Enterprise services

Apart from transferring from multihomed personal networks to multihomed PA enterprise setups without any changes, destination-source routing can also be used to correctly route services that assign their own prefixes to customers using the particular service. This is distinct from internet connectivity only in that it does not provide a default route. Applying destination-source routing, the entire routing domain is aware of the specific constraints of the routes involved.

Additionally, if the walled-garden's destination prefix is advertised as blackhole route, this ensures that communication with the service will only be routed using the specific D/S route, never leaking onto unintended paths like a default route.

This is very similar to firewall/filtering functionality, except the feature is distributed onto routers.

2.5. Information Source for Neighbor Management

Having information on source address restrictions for routes distributed, routers can rely on this additional information to improve their behaviour towards hosts connected to them. This specifically includes IPv6 Router Advertisements, which is described in [RFC8028] and [RFC8475].

3. Principle of operation

3.1. Frame of reference

The principles described here are define on a functional level what the semantics of routing information exchanged between systems is. It is neither a prescription in how to efficiently implement these semantics, nor does it preclude an implementation from providing other administrator-friendly views of the same routing information.

More specifically, forwarding plane implementations are expected to internally diverge from the lookup algorithm described below. The router as a whole MUST ultimately behave as if the steps below were followed. An internal variation providing improved performance, as well as a variation matching existing implementations with reversed order are described in Appendix A.1 and Appendix A.2, respectively.

3.2. Route information and equality

The mechanism in this document is such that a source prefix is added to all route entries. This document assumes all entries have a source prefix, with `::/0` as default value for entries installed without a specified source prefix. This need not be implemented in this particular way, however the system MUST behave exactly as if it were. In particular, a difference in behaviour between routes with a source prefix of `::/0` and routes without source prefix MUST NOT be visible.

For uniqueness considerations, the source prefix factors MUST be taken into account for comparisons. Two routes with identical information except the source prefix MAY exist and MUST be installed and matched.

3.3. Lookup ordering and disambiguation

When a router is making packet forwarding decision, that is consulting its routing table in order to determine next-hop to forward the packet to, it will use information from packet's header to look up best matching route from the routing table. This section describes lookup into the destination-source routing table.

For longest-match lookups, the source prefix is matched after the destination prefix. This is to say, first the longest matching destination prefix is found, then the table is searched for the route with the longest source prefix match, while only considering routes with exactly the destination prefix previously found. If and only if no such route exists (because none of the source prefixes match), the lookup moves to the next less specific destination prefix.

A router MUST continue to a less specific destination prefix if no route matches on the source prefix. It MUST NOT terminate lookup on such an event.

Using $A \subset B$ to mean "A is more specific than B", this is represented as:

$$A \subset B := \begin{array}{l} \text{Adst} \subset \text{Bdst} \\ \vee (\text{Adst} == \text{Bdst} \ \&\& \ \text{Asrc} \subset \text{Bsrc}) \end{array}$$

```
// RFC editor note (1) (to be removed before publishing): this
// originally used < rather than C. Using < as a symbol is
// _slightly_ ambiguous due to the neverending semiotic conflict with
// a _larger_ prefix having a _shorter_ prefix length. There is no
// strong need to use C here, reverting to < (and thus ASCII) is also
// fine.
```

```
// RFC editor note (2) (to be removed before publishing): The two
// lines above are carefully formatted to vertically align their
// terms. Setting it as figure/artwork/monospaced code block gets
// this effect in TXT and HTML outputs, but not PDF. Any efforts to
// improve this formatting are much appreciated.
```

3.4. Ordering Rationale

Ordering of searching for address match is important and reversing it would lead to semantically different behavior. This standard requires most specific match on destination address to be found before looking for match on source address.

Choosing destination to be evaluated first caters to the assumption that local networks should have full, contiguous connectivity to each other. This implies that those specific local routes always match first based on destination, and use a zero ("all sources") source prefix.

If the source prefix were to be matched first, this would result in a less specific (e.g. default) route with a source prefix to match before those local routes. In other terms, this would essentially divide local connectivity into zones based on source prefix, which is not the intention of this document.

Hence, this document describes destination-first match search.

4. Routing protocol considerations

As with the destination-only routing, destination-source routes will typically be disseminated throughout the network by dynamic routing protocols. It is expected that multiple dynamic routing protocols will be adapted to the needs of destination-source routing architecture. Specification of dynamic routing protocols is outside of scope of this document. This section lists requirements and considerations for the dynamic destination-source routing protocols.

4.1. Source information

Dynamic routing protocols will need to be able to propagate source range information together with destination prefix and other accompanying routing information. Source range information may be propagated with all destination prefixes or only some of them. Destination prefixes advertised without associated source range MUST be treated as having default source range `::/0`.

Dynamic routing protocols MUST be able to propagate multiple routes whose destination prefix is the same but associated source ranges are different. Such unique pairs of destination and source MUST be treated as different destination-source routes.

There is no limitation on how source range information is propagated and associated with destination prefixes. Individual protocols may choose to propagate source range together with a destination prefix in the form of prefix, in the form of index to list of known source ranges or in any other form allowing receiver to reconstruct pair of destination prefix and associated source range.

4.2. Loop-freeness considerations

It is expected that some existing dynamic routing protocols will be enhanced to propagate destination-source routing information. In this case the protocol may be configured to operate in a network where some, but not all, routers support destination-source routing and others are still using destination-only routing. Even if all routers within a network are capable of destination-source routing, it is very likely that on edges of the network they will have to forward packets to routers doing destination-only routing.

Since a router implementing destination-source routing can have additional, more granular routes than one that doesn't implement it, persistent loops can form between these systems.

Thus specifications of destination-source routing protocols (either newly defined protocols or enhancements to already existing one) MUST take provisions to guarantee loop-free operations.

There are 3 possible approaches to avoid looping condition:

1. Guarantee that next-hop gateway of a destination-source route supports destination-source routing, for example calculate an alternate topology including only routers that support destination-source routing architecture

2. If next-hop gateway is not aware of destination-source routing then a destination-source path can lead to it only if next-hop router is 'closer' to the destination in terms of protocol's routing metric; important particular case of the rule is if destination-only routing is pointing to the same next-hop gateway
3. Discard the packet (i.e. treat destination-source route as unreachable)

In many practical cases routing information on the edges of destination-source routing domain will be provided by an operator via configuration. Dynamic routing protocol will only disseminate this trusted external routing information. For example, returning to the use case of multihomed Home network (Section 2.1), both routers R1 and R2 will have default static routes pointing to ISPs.

Above considerations require a knowledge of the next-hop router's capabilities. For routing protocols based on hop-by-hop flooding (RIP [RFC2080], BGP [RFC4271]), knowing the peer's capabilities is sufficient. Information about if peer supports destination-source routing can either be negotiated explicitly or simply be deduced from the fact that systems would propagate destination-source routing information only if they understand it. Protocols building a link-state database (OSPFv3 [RFC5340], IS-IS [RFC5308]) have the additional opportunity to calculate alternate paths based on knowledge of the entire domain but cannot assume that routers understand destination-source routing information only because they participated in its flooding. Such protocols **MUST** explicitly advertise support for the destination-source routing.

4.3. Recursive routing

Dynamic routing protocols may propagate routing information in a recursive way. Examples of such recursion is forwarding address in OSPFv3 [RFC5340] AS-External-LSAs and NEXT_HOP attribute in BGP [RFC4271] NLRI.

Dynamic routing protocol supporting recursive routes **MUST** specify how this recursive routing information is interpreted in the context of destination-source routing as part of standardizing destination-source routing extensions for the protocol. Section 5.1 lists several possible strategies protocols can choose from.

5. Applicability To Specific Situations

This section discusses how destination-source routing is used together with some common networking techniques dependent on routes in the routing table.

5.1. Recursive Route Lookups

Recursive routes provide indirect path information where instead of supplying the next-hop directly they specify that next-hop information must be taken from another route in the same routing table. It is said that one route 'recurses' via another route which is 'resolving' recursion. Recursive routes may either be carried by dynamic routing protocols or provided via configuration as recursive static routes.

Recursive destination-source routes have additional complication in how source address range should be considered while finding destination-source route to resolve recursion.

There are several possible approaches:

1. Ignore destination-source routes, resolve recursion only via destination-only routes (i.e. routes with source range `::/0`)
2. Require that both the recursive and resolving routes have the same source range associated with them; this requirement may be too restrictive to be useful in many cases
3. Require that source range associated with recursive route is a subset of source range associated with route resolving recursion (i.e. source range of the resolving route is less specific superset of recursive route's source range)
4. Create multiple instances of the route whose nexthop is being resolved with different source prefixes; this option is further elaborated in Section 5.1.1

When recursive routing information is propagated in a dynamic routing protocol, it is up to the protocol specification to select and standardize appropriate scheme of recursive resolution.

Recursive resolution of configured static routes is local to router where recursive static routes were configured, thus behavior is implementation's choice. Implementations SHOULD provide option (3) from the above list as their default method of recursive static route resolution. This is both to guarantee that destination-only recursive static routes do not change their behavior when router's software is upgraded to support destination-source routing and at the same time make destination-source recursive routes useful.

5.1.1. Recursive route expansion

When doing recursive nexthop resolution, the route that is being resolved is installed in potentially multiple copies, inheriting all possible more-specific routes that match the nexthop as destination. The algorithm to do this is:

1. form the set of attributes for lookup by using the (unresolved, recursive) nexthop as destination (with full host prefix length, i.e. /128), copy all other attributes from the original route
2. find all routes that overlap with this set of attributes (including both more-specific and less-specific routes)
3. order the result from most to less specific
4. for each route, install a route using the original route's destination and the "logical and" overlap of each extra match attribute with same attribute from the set. Copy nexthop data from the route under iteration. Then, reduce the set of extra attributes by what was covered by the route just installed ("logical AND NOT").

Example recursive route resolution

route to be resolved:

```
2001:db8:1234::/48, source 2001:db8:3456::/48,  
    recursive nexthop via 2001:db8:abcd::1
```

routes considered for recursive nexthop:

```
::/0,                                via fe80::1%1  
2001:db8:abcd::/48,                  via fe80::2%2  
2001:db8:abcd::/48,    source 2001:db8:3456:3::/64, via fe80::3%3  
2001:db8:abcd::1/128, source 2001:db8:3456:4::/64, via fe80::4%4
```

recursive resolution result:

```
2001:db8:1234::/48,    source 2001:db8:3456::/48,    via fe80::2%2  
2001:db8:1234::/48,    source 2001:db8:3456:3::/64, via fe80::3%3  
2001:db8:1234::/48,    source 2001:db8:3456:4::/64, via fe80::4%4
```

5.2. Unicast Reverse Path Filtering

Unicast reverse path filtering MUST use dst-src routes analog to its usage of destination-only routes. However, the system MAY match either only incoming source against routes' destinations, or it MAY match source and destination against routes' destination and source. It MUST NOT ignore dst-src routes on uRPF checks.

5.3. Multicast Reverse Path Forwarding

Multicast Reverse Path Lookups are used to find paths towards the (known) sender of multicast packets. Since the destination of these packets is the multicast group, it cannot be matched against the source part of a dst-src route. Therefore, dst-src routes MUST be ignored for Multicast RPF lookups.

5.4. Testing for Connectivity Availability

There are situations where systems' behaviour depends on the fact whether "connectivity" is available in a broad sense. These systems may have previously tested for the existence of a default route in the routing table.

Since the default route may now be qualified with a source prefix, this test can fail. If no additional information is available to qualify this test, systems SHOULD test for the existence of any default route instead, e.g. include routes with default destination but non-default source prefix.

However, if the test can be associated with a source address or source prefix, this data SHOULD be used in looking up a default route. Depending on the application, it MAY also be useful to - possibly additionally - consider "connectivity" to be available if any route exists where the route's source prefix covers the prefix or address under consideration, allowing arbitrary destination prefixes.

Note though that this approach to routing SHOULD NOT be used to infer a list of source prefixes in an enumerative manner, or even to guess domain information. Specifically, if an operator uses more specific source prefixes to refine their routing, the inferred information will provide bogus extraneous output. This is distinct from the connectivity tests mentioned above in that those actually inquire the routing system, unlike domain information or enumeration, which is higher-layer application information.

5.5. Performing Source Address Selection

Any system implementing destination-source routing, including plain end hosts, needs some way to select a source address particularly for outgoing connections. While the application establishing the connection can take over this function (i.e. binding a socket to a source address), this is generally performed by the system following [RFC6724].

Critically, selecting a source address is done after looking up a route to the destination and Section 5 of [RFC6724] makes multiple references to the path "used to send to D". This means that implementations MUST provide some manner of establishing useful results for this lookup, done without known source address.

Details of what exact result this routing lookup yields or how it is implemented is essentially analogous to choosing between equal candidate routes (i.e. distinct routes, not one route with multiple nexthops) in a configuration that does not implement destination-source routing. The result is a matter and expression of local policy.

It is RECOMMENDED to choose one of the following options:

- * an implementation MAY, as necessary, install destination-source routes with a source prefix of `::/128`, and perform routing lookups for the purpose of source address selection by using that source prefix.
- * an implementation MAY introduce and use a separate route lookup function (that does not use destination-source routing) for the purpose of source address selection

Above list is not exhaustive and other approaches may be valid and useful depending on context.

5.6. Improving Source Address Selection

Beyond the considerations in the previous section, source address selection and routing choice is key to improving host behavior in the presence of multiple routers, prefixes or interfaces. Rule 5.5 as noted in Section 5 of [RFC6724], and [RFC8028] (note [RFC8028-ERRATA7009]) are at the heart of this.

Unfortunately, the functionality described by the text in rule 5.5 is insufficient to make these scenarios work. A source address is chosen in context of a particular nexthop, but there is no mechanism preferring that nexthop for packets exchanged afterwards. [RFC8028] is intended to address this and makes several recommendations involving use of the source address.

Following these recommendation and resolving the issues resulting from it essentially yields destination-source routing. The more general the solution is, the closer it looks and functions to destination-source routing.

Destination-source routing is explicitly applicable to hosts, and in the context of fully implementing [RFC6724] and [RFC8028], it is RECOMMENDED that end hosts in fact implement destination-source routing.

Additional routes (with source prefixes) need to be installed into the routing table in the course of this. The exact semantics of this are left to be specified in a future document.

6. Interoperability

As pointed out in Section 4.2 traffic may permanently loop between routers forwarding packets based only on their destination IP address and routers using both source and destination addresses for forwarding decision.

In networks where the same dynamic routing protocol is being used to propagate routing information between both types of systems the protocol may address some or all traffic looping problems. Recommendations to protocol designers are discussed in Section 4.2.

When routing information is coming from outside of the routing protocol (for example, being provided by operator in the form of static routes or network protocols not aware of destination-source routing paradigm) it may not be possible for the router to ascertain loop-free properties of such routing information. In these cases consistent (and loop-free) packet forwarding is woven into network topology and must be taken into consideration at design time.

It is possible to design network with mixed deployment of routers supporting and not supporting destination-source routing. Thus gradual enablement of destination-source routing in existing networks is also possible but has to be carefully planned and evaluated for each network design individually.

Generally, destination-source routing will not cause traffic loops when disjoint 'islands' of destination-source routing do not exchange destination-source routing information. One particular case of this rule is a network which contains single contiguous 'island' of routers aware of destination-source routing. Example SOHO network from Section 2.1 which demonstrates this design approach:

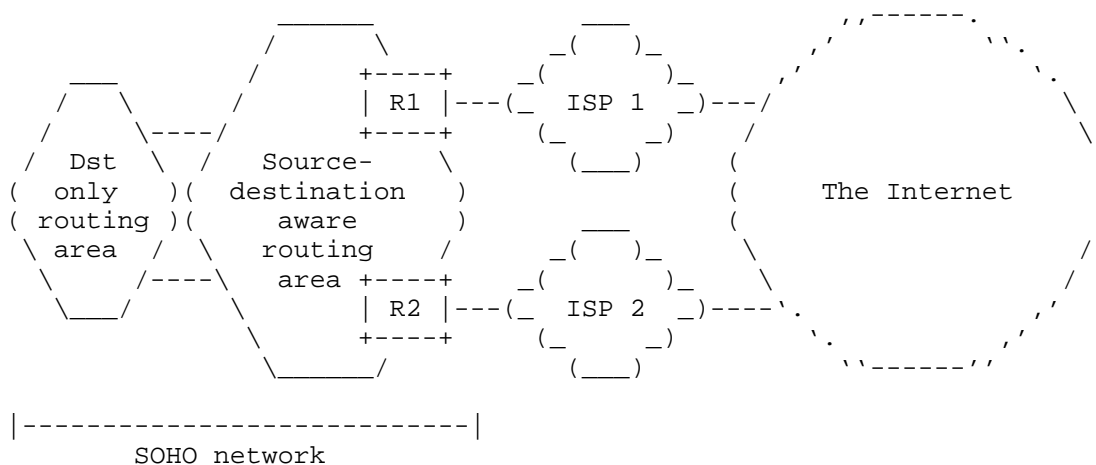


Figure 2: Example of multihomed small network with partial deployment of destination-source routing

6.1. Interoperability in Distance-Vector Protocols

Distance-Vector routing protocols (BGP, RIPng, BABEL), operating on a hop-by-hop basis, can address interoperability and migration concerns on that level. With routing information being flooded in the reverse direction of traffic being forwarded using that information, a hop that floods is the same hop that forwards.

This makes dealing with destination/source-unaware routers easy if destination/source routes are made to be ignored by such unaware routers, and flooding of such routes is inhibited.

If D/S routes are discarded by non-D/S routers, D/S routers will not receive non-working routes and can select from other available working D/S routes.

Note that for this to work, non-D/S routers MUST NOT flood D/S routing information. This can be achieved in 2 ways:

1. Using some preexisting encoding to signal non-D/S routers to not flood these particular routes
2. Ignoring flooded D/S information on D/S routers by having them detect that they received it from a non-D/S router (e.g. using some capability signalling to identify non-D/S routers.) This handling likely needs to be performed on a level of same-link neighborships.

Also note that the considerations in this section only apply if data path and flooding path are congruent.

6.2. Interoperability in Link-State Protocols

For Link-State routing protocols (OSPF, IS-IS), there is no relation between route flooding and forwarding. Instead, forwarding decisions are based on shortest-path calculation on top of the received topology information.

For a D/S router to avoid loops, there are again two choices available:

1. Detect that forwarding for a D/S route transits over a non-D/S router and convert the route into a blackhole route to replace looping with blackholing. This obviously impacts connectivity.
2. Perform separate SPF calculations using only the subset of D/S-capable routers; thus D/S routers can forward D/S-routed packets as long as they stay in contiguous islands.

The latter approach is facilitated by Multi-Topology extensions to the respective protocols. These extensions provide a way to both isolate D/S routing information and perform the separate SPF calculation. Note that it is not necessary to use multiple topologies for distinct source prefixes; only a single additional topology encompassing all D/S-capable routers is sufficient.

7. IANA Considerations

This document makes no requests to IANA.

8. Security Considerations

Systems operating under the principles of this document can have routes that are more specific than the previously most specific, i.e. host routes. This can be a security concern if an operator was relying on the impossibility of hijacking such a route.

While destination-source routing could be used as part of a security solution, it is not really intended for the purpose. The approach limits routing, in the sense that it routes traffic to an appropriate egress, or gives a way to prevent communication between systems not included in a destination-source route, and in that sense could be considered similar to an access list that is managed by and scales with routing.

9. Privacy Considerations

If a host's addresses are known, injecting a dst-src route allows isolation of traffic from that host, which may compromise privacy. However, this requires access to the routing system. As with similar problems with the destination only, defending against it is left to general mechanisms protecting the routing infrastructure.

10. Acknowledgements

The base underlying this document was first outlaid by Ole Troan and Lorenzo Colitti in [I-D.troan-homenet-sadr] for application in the homenet area. Significant contributions to source-specific routing as a whole came from Juliusz Chroboczek and Matthieu Boutier. Matthieu has also provided a huge amount of review and editing input on this document.

This document itself is largely the result of discussions with Fred Baker and derives from [I-D.baker-ipv6-isis-dst-src-routing].

Thanks to Chris Bowers, Jeffrey Haas, Ted Lemon, Acee Lindem, Tony Przygienda, and ric Vyncke for their input and review.

The Linux kernel is providing an implementation of the behaviour described here since even before the document was started.

11. Change Log

September 2025 [-04]: no content changes (author metadata only)

March 2025 [-03]: 2 new sections about source address selection
editorial fixes, updated references

October 2024 [-02]: no changes

July 2024 [-01]: addressed comments during WG adoption call

July 2024 [-00]: renamed to draft-ietf-rtgwg-dst-src-routing-revive,
no content changes

July 2023 [-01]: added section on implementation status

July 2022 [-00]: renamed to draft-llsyang-rtgwg-dst-src-routing,
no content changes

March 2019 [-07]: no changes

October 2017 [-06]: clarify described operation is not an implementation guide

editorial cleanups

July 2017 [-05]: clarify connectivity tests

extend use cases

editorial cleanups

May 2017 [-04]: no changes

November 2016 [-03]: added DV/LS protocol considerations

note backtracking workaround/caveat

November 2015 [-02]: added section on destination-source routing use cases

added section on alternative lookup algorithm

added section on requirement for dynamic routing protocols
dissiminating destination-source information

October 2015 [-00]: renamed to draft-ietf-rtgwg-dst-src-routing-00,
no content changes from draft-lamparter-rtgwg-dst-src-routing-01.

April 2015 [-01]: merged routing-extra-qualifiers draft, new
ordering rationale section

October 2014 [-00]: Initial Version

12. References

12.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

- [RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <<https://www.rfc-editor.org/info/rfc8200>>.

12.2. Informative References

- [hal-00947234v1] Boutier, M. and J. Chroboczek, "Source-sensitive routing", hal 00947234v1, 2014, <<https://hal-univ-diderot.archives-ouvertes.fr/hal-00947234v1>>.
- [I-D.baker-ipv6-isis-dst-src-routing] Baker, F. and D. Lamparter, "IPv6 Source/Destination Routing using IS-IS", Work in Progress, Internet-Draft, draft-baker-ipv6-isis-dst-src-routing-07, 18 July 2017, <<https://datatracker.ietf.org/doc/html/draft-baker-ipv6-isis-dst-src-routing-07>>.
- [I-D.troan-homenet-sadr] Tran, O. and L. Colitti, "IPv6 Multihoming with Source Address Dependent Routing (SADR)", Work in Progress, Internet-Draft, draft-troan-homenet-sadr-01, 4 September 2013, <<https://datatracker.ietf.org/doc/html/draft-troan-homenet-sadr-01>>.
- [RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, DOI 10.17487/RFC0791, September 1981, <<https://www.rfc-editor.org/info/rfc791>>.
- [RFC2080] Malkin, G. and R. Minnear, "RIPng for IPv6", RFC 2080, DOI 10.17487/RFC2080, January 1997, <<https://www.rfc-editor.org/info/rfc2080>>.
- [RFC2827] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", BCP 38, RFC 2827, DOI 10.17487/RFC2827, May 2000, <<https://www.rfc-editor.org/info/rfc2827>>.
- [RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <<https://www.rfc-editor.org/info/rfc4271>>.
- [RFC5308] Hopps, C., "Routing IPv6 with IS-IS", RFC 5308, DOI 10.17487/RFC5308, October 2008, <<https://www.rfc-editor.org/info/rfc5308>>.

- [RFC5340] Coltun, R., Ferguson, D., Moy, J., and A. Lindem, "OSPF for IPv6", RFC 5340, DOI 10.17487/RFC5340, July 2008, <<https://www.rfc-editor.org/info/rfc5340>>.
- [RFC6724] Thaler, D., Ed., Draves, R., Matsumoto, A., and T. Chown, "Default Address Selection for Internet Protocol Version 6 (IPv6)", RFC 6724, DOI 10.17487/RFC6724, September 2012, <<https://www.rfc-editor.org/info/rfc6724>>.
- [RFC8028] Baker, F. and B. Carpenter, "First-Hop Router Selection by Hosts in a Multi-Prefix Network", RFC 8028, DOI 10.17487/RFC8028, November 2016, <<https://www.rfc-editor.org/info/rfc8028>>.
- [RFC8028-ERRATA7009]
Carpenter, B., "RFC Errata Report #7009, on RFC8028 (First-Hop Router Selection by Hosts in a Multi-Prefix Network)", June 2022, <<https://www.rfc-editor.org/errata/eid7009>>.
- [RFC8043] Sarikaya, B. and M. Boucadair, "Source-Address-Dependent Routing and Source Address Selection for IPv6 Hosts: Overview of the Problem Space", RFC 8043, DOI 10.17487/RFC8043, January 2017, <<https://www.rfc-editor.org/info/rfc8043>>.
- [RFC8475] Linkova, J. and M. Stucchi, "Using Conditional Router Advertisements for Enterprise Multihoming", RFC 8475, DOI 10.17487/RFC8475, October 2018, <<https://www.rfc-editor.org/info/rfc8475>>.

Appendix A. Implementation Options

A.1. Pre-expanded 2-step lookup without backtracking

The backtracking behavior (specified in Section 3.3 as "A router MUST continue to a less specific destination prefix") has been shown to potentially cause a significant loss of forwarding performance since forwarding a single packet may require a large number of table lookups. (The degenerate case is 129 destination lookups in decreasing prefix length, each followed by a failing longest-match on the source prefix.)

To avoid this, implementations can install synthetic routes to achieve the same lookup result. This works as follows, to be evaluated for each unique destination prefix:

1. If there is a route (D, S::

2. Iterate upwards one level (from D if first iteration, previous D' otherwise) to a less specific destination. Call this D'.
3. For all routes (D', S'), i.e. all source prefixes S' under that destination prefix, install a copy (D, S') if and only if S' covers some source prefix that isn't covered yet. (In terms of set theory, S' cut by all existing S under D is not empty.)
4. Repeat at step 1.

The effect of this algorithm is that after performing a lookup on the destination prefix, looking up the source prefix directly yields the result that backtracking would give. This eliminates backtracking and provides constant 2 lookup cost (after exactly one destination longest-match, the source longest-match will provide the final, correct result; any no-match is a final no-match).

A.2. Translation to Multi-FIB (Policy Routing) perspective

The lookup procedure described in this document requires destination-first lookup. This is not a fit with most existing implementations of Policy Routing. While Policy Routing has no formal specification, it generally permits choosing from multiple routing tables / FIBs based on, among other things, source address. Some implementations support using more than one FIB for a single lookup, but not all do.

An implementation that can choose from multiple FIBs based on source address is capable of correct forwarding according to this document, provided that it supports enough FIBs. One FIB will be used for each unique source prefix.

For a complete description of the required translation algorithm, please refer to [hal-00947234v1]. It roughly works as follows:

After destination-source routing information has been collected, one FIB table is created for each source range including the default range `::/0`. Source-destination routes then replicated into each destination-only FIB table whose associated source address range is a subset of route's source range. Note that this rule means routes with default source range `::/0` are replicated into each FIB table.

In case when multiple routes with the same destination prefix are replicated into the same FIB table only route with the most specific source address range is installed.

For example, if destination-source routing table contains these routes:

Destination prefix	Source range	Next Hop
::/0,	::/0,	NH1
2001:101:1234::/48,	2001:db8:3456:8000::/56,	NH2
2001:101:5678::/48,	2001:db8:3456:8000::/56,	NH3
	::/0,	NH4
2001:101:abcd::/48,	2001:db8:3456::/48,	NH5

Table 1

then 3 FIB tables will be created associated with source ranges ::/0, 2001:db8:3456::/48 and 2001:db8:3456:8000::/56. In this example range 2001:db8:3456:8000::/56 is a subset of less specific range 2001:db8:3456::/48. Such inclusion makes a somewhat artificial example but was intentionally selected to demonstrate hierarchy of route replication.

And content of these FIB tables will be:

FIB 1 (source range ::/0):

Destination prefix	Next Hop
::/0,	NH1
2001:101:5678::/48,	NH4

Table 2

FIB 2 (source range 2001:db8:3456::/48):

Destination prefix	Next Hop
::/0,	NH1
2001:101:5678::/48,	NH4
2001:101:abcd::/48,	NH5

Table 3

FIB 3 (source range 2001:db8:3456:8000::/56):

Destination prefix	Next Hop
::/0,	NH1
2001:101:1234::/48,	NH2
2001:101:5678::/48,	NH3
2001:101:abcd::/48,	NH5

Table 4

During packet forwarding, lookup first matches source address against the list of address ranges associated with FIB tables to select a FIB table with the most specific source address range and then does destination-only lookup in the selected FIB table.

Appendix B. Implementation Status

This section is to be removed before publishing as an RFC.

The Linux kernel implementation of destination-source routing - which predates pretty much all other work on this - is cited in the acknowledgements section. The CONFIG_IPV6_SUBTREES setting enables the code.

The destination/source routing has been implemented and tested under an experiment network with four routers that supports both the new routing and forwarding plane. China Education and Research Network (CERNET) has deployed 20 upgraded destination/source routers recently.

Authors' Addresses

David 'equinox' Lamparter
NetDEF, Inc.
Leipzig
Germany
Email: equinox@diac24.net, equinox@opensourcerouting.org

Anton Smirnov
Cisco Systems, Inc.
De Kleetlaan 6a
1831 Diegem
Belgium
Email: as@cisco.com

Jen Linkova
Google
1 Darling Island Rd
Pyrmont NSW 2009
Australia
Email: furryl3@gmail.com, furry@google.com

Shu Yang (editor)
Shenzhen University
Shenzhen
518060
China
Email: yang.shu@szu.edu.cn

Mingwei Xu
Tsinghua University
Beijing
100084
China
Email: xumw@tsinghua.edu.cn