Network Working Group M. Boutier
Internet-Draft J. Chroboczek
Updates: 6126bis (if approved) IRIF, University of Paris-Diderot
Intended status: Standards Track August 21, 2017
Expires: February 22, 2018

Source-Specific Routing in Babel
draft-ietf-babel-source-specific-01

Abstract

Source-specific routing is an extension to traditional next-hop routing where packets are forwarded according to both their destination and their source address. This document describes the source-specific routing extension to the Standard Track's Babel routing protocol defined in [BABEL]. It is incompatible with the Experimental Track's Babel [RFC6126].

Source-specific routing is also known as Source Address Dependent Routing, SAD Routing, SADR, Destination/Source Routing or Source/Destination Routing.

Table of Contents

1. TODOs 2
2. Introduction and background 2
3. Data Structures 3
  3.1. The Source Table 3
  3.2. The Route Table 4
  3.3. The Table of Pending Seqno Requests 4
4. Data Forwarding 4
5. Protocol Operation 5
  5.1. Source-specific messages 6
  5.2. Route Acquisition 6
  5.3. Wildcard retractions (update) 6
  5.4. Wildcard requests 6
6. Compatibility with the base protocol 7
  6.1. Loop-avoidance 7
  6.2. Starvation and Blackholes 8
7. Protocol Encoding 8
  7.1. Source Prefix sub-TLV 8
  7.2. Source-specific Update 9
  7.3. Source-specific (Route) Request 9
  7.4. Source-Specific Seqno Request 9
8. IANA Considerations 9
9. Security considerations 9
10. References 10
  10.1. Normative References 10
  10.2. Informative References 10
Authors' Addresses 10

1. TODOs

- o Source Prefix sub-TLV type: TBD
- o check references (Section) for BABEL in 6126bis

2. Introduction and background

The Babel routing protocol as defined is [BABEL] is a distance vector routing protocol for next-hop routing. In next-hop routing, each node maintains a forwarding table which maps prefixes to next-hops. The forwarding decision is a per-packet operation which depends on

the destination address of the packets and on the entries of the forwarding table. When a packet is about to be routed, its destination address is compared to the prefixes of the routing table: the entry with the most specific prefix containing the destination address of the packet is choosen, and the packet is forwarded to the associated next-hop. Next-hop routing is a simple, well understood paradigm that works satisfactorily in a large number of cases.

Source-specific routing is a modest extension of next-hop routing where the forwarding decision additionnaly depends on the source address of the packets. The forwarding tables are extended to map pairs of prefixes (destination, source) to a next-hop. When multiple entries are candidate to route a packet, the one with the most specific destination prefix is choosen, and in case of equality the one with the most specific source. In source-specific routing, two packets with the same destination but different sources may be forwarded among different paths.

The main application of source-specific routing is, at the time of this writing, multihoming with Provider Agregatable (PA) addresses. In such configuration, each Internet Service Provider (ISP) provides to the network a PA prefix and a default route for this prefix while performing ingress filtering ([BCP84]). Each host has one address per ISP, and sends packets with one of these addresses as source address. Source-specific routing ensures that packets are routed towards the provider of their source address, such that they are not filtered out. More details and more use cases can be found in [SS-ROUTING],[IETF-SSR].

This document describes the source-specific routing extension for the Babel routing protocol [BABEL]. This involves changes to data structures and protocol messages. The data structures receive an additionnal source prefix which is part of the index, similarly to (and with) the destination prefix. The Update, Route Request and Seqno Request are the three messages which carry a (destination) prefix: they are extended with a source prefix.

3. Data Structures

Some of the data structures of a Babel node contains a destination prefix or are partly indexed by a destination prefix. This extension adds a source prefix to these structures and indexes.

3.1. The Source Table

Every Babel node maintains a source table, as described in [BABEL], Section 3.2.5. A source-specific Babel node extends this table with

the following field. With this extension, the source table is indexed by triples of the form (prefix, source prefix, router-id).

o the source prefix specifying the source address of packets to which this entry applies.

If a source table entry has a zero length source prefix, then the entry is a non-source-specific entry, and is treated just like a source table entry defined by the original Babel protocol.

With this extension, the route entry contains a source which itself contains a source prefix. These are two very different concepts, and should not be confused.

3.2. The Route Table

Every Babel node maintains a route table, as described in [BABEL], Section 3.2.6. With this extension, the route table is indexed by triples of the form (prefix, source prefix, neighbour) obtained from the associated source table entry.

If a route table entry has a zero length source prefix, then the entry is a non-source-specific entry, and is treated just like a route table entry defined by the original Babel protocol.

3.3. The Table of Pending Seqno Requests

Every Babel node maintains a table of pending seqno requests, as described in [BABEL], Section 3.2.7. A source-specific Babel node extends this table with the following entry. With this extension, the table of pending seqno requests is indexed by triples of the form (prefix, source prefix, router-id).

o the source prefix being requested.

4. Data Forwarding

In next-hop routing, if two routing table entries overlap, then one is necessarily more specific than the other; the "longest prefix rule" specifies that the most specific applicable routing table entry is chosen.

With source-specific routing, there might no longer be a most specific applicable entry: two routing table entries might match a given packet without one necessarily being more specific than the other. Consider for example the following routing table:

destination	source	next-hop
2001:DB8:0:1::/64	::/0	A
::/0	2001:DB8:0:2::/64	B

This specifies that all packets with destination in 2001:DB8:0:1::/64 are to be routed through A, while all packets with source in 2001:DB8:0:2::/64 are to be routed through B. A packet with source 2001:DB8:0:2::42 and destination 2001:DB8:0:1::57 matches both rules, although neither is more specific than the other. A choice is necessary, and unless the choice being made is the same on all routers in a routing domain, persistent routing loops may occur. More informations are available in [SS-ROUTING] Section IV.C.

A Babel implementation MUST choose routing table entries by using the so-called destination-first ordering, where a routing table entry R1 is preferred to a routing table entry R2 when either R1's destination prefix is more specific than R2's, or the destination prefixes are equal and R1's source prefix is more specific than R2's. (In more formal terms, routing table entries are compared using the lexicographic product of the destination prefix ordering by the source prefix ordering.)

In practice, this means that a source-specific Babel implementation must take care that any lower layer that performs packet forwarding obey this semantics. In particular:

- If the lower layers implement the destination-first ordering, then the Babel implementation MAY use them directly;

- If the lower layers can hold source-specific routes, but not with the right semantics, then the Babel implementation MUST disambiguate the routing table by using a suitable disambiguation algorithm (see [SS-ROUTING] Section V.B for such an algorithm);

- If the lower layers cannot hold source-specific routes, then a Babel implementation MUST silently ignore (drop) any source-specific routes.

5. Protocol Operation

This extension does not fundamentally change the operation of the Babel protocol. We only describe the fundamental differences between the original protocol and this extension in this section. The other mechanisms described in [BABEL] (Section 3) are extended to pairs of (destination, source) prefixes instead of just (destination) prefixes.

5.1. Source-specific messages

Three messages carry a destination prefix: Updates, Route Requests and Seqno Requests. These messages are extended to carry, in addition, a source prefix if (and only if) the corresponding route is source-specific. More formally, an Update, a Route Request and a Seqno Request MUST carry a source prefix if they concern a source-specific route (non-zero length source prefix) and MUST NOT carry a source prefix otherwise (zero length source prefix). A message which carries a source prefix is said to be source-specific.

5.2. Route Acquisition

When a non-source-specific Babel node receives a source-specific update, it silently ignores it. When a source-specific Babel node receives a non-source-specific update, it MUST treat this update as a zero length source-specific update.

When a source-specific Babel node receives a source-specific update (prefix, source prefix, router-id, seqno, metric) from a neighbour neigh, it behaves as described in [BABEL] (Section 3.5.4) though indexing entries by (prefix, source prefix, neigh).

5.3. Wildcard retractions (update)

The original protocol defines a wildcard update with AE equals to 0 as being a wildcard retraction. A node receiving a wildcard retraction on an interface must consider that the sending node retracts all the routes it advertised on this interface.

Wildcard retractions are used when a node is about to leave the network. Thus, this extension does not define source-specific wildcard retraction, but extends wildcard retraction to apply also to source-specific routes. More formally, a wildcard update MUST NOT carry a source prefix, and a source-specific Babel node receiving a (legacy) wildcard update MUST retracts all routes it learns from this node (including source-specific ones).

5.4. Wildcard requests

The original Babel protocol states that when a node receives a wildcard route request, it SHOULD send a full routing table dump. This extension does not change this statement: a source-specific node SHOULD send a full routing table dump when receiving a wildcard request.

Source-specific wildcard requests does not exist: a wildcard request MUST NOT carry a source prefix, and a source prefix associated with a wildcard update SHOULD be ignored.

One of the motivation behalf this design choice is that wildcard requests are defined with AE equals to 0. They naturally apply to AE 1, AE 2 and AE 3 defined in [BABEL], but also to any other AE which may be defined in the future. New AEs, new TLVs or new sub-TLVs are extension mechanisms. Thus, the semantics of a wildcard request is clearly to also asks for routes coming from extensions.

6. Compatibility with the base protocol

The protocol extension defined in this document is, to a great extent, interoperable with the base protocol defined in [BABEL] (and all its known extensions). More precisely, if non-source-specific routers and source-specific routers are mixed in a single routing domain, Babel's loop-avoidance properties are preserved, and, in particular, no persistent routing loops will occur.

However, this extension is not compatible with the Experimental Track's Babel Routing Protocol [RFC6126]. It requires the mandatory sub-TLV introduced in [BABEL]. Consequently, this extension MUST NOT be used with routers implementing RFC 6126, otherwise persistent routing loops may occur.

6.1. Loop-avoidance

The extension defined in this protocol uses a new Mandatory sub-TLV to carry the source prefix information. As discussed in Section 4.4 of [BABEL], this encoding ensures that non-source-specific routers will silently ignore the whole TLV, which is necessary to avoid persistent routing loops in hybrid networks.

Consider two nodes A and B, with A source-specific announcing a route to (D, S). Suppose that B (non source-specific) merely ignores the source prefix information when it receives the update rather than ignoring the whole TLV, and reannounces the route as D. This reannouncement reaches A, which treats it as (D, ::/0). Packets destined to D but not sourced in S will be forwarded by A to B, and by B to A, causing a persistent routing loop:

```
    (D,S)                 (D)
     <--                 <--
  ------ A ----------------- B
           -->
          (D,::/0)
```

6.2. Starvation and Blackholes

In general, discarding source-specific routes by non-source-specific routers will cause route starvation. Intuitively, unless there are enough non-source-specific routes in the network, non-source-specific routers will suffer starvation, and discard packets for destinations that are only announced by source-specific routers.

A simple yet sufficient condition for avoiding starvation is to build a connected source-specific backbone that includes all of the edge routers, and announce a (non-source-specific) default route towards the backbone.

7. Protocol Encoding

This extension defines a new sub-TLV used to carry a source prefix by the three following existing messages: Update, Route Request and Seqno Request.

7.1. Source Prefix sub-TLV

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Type = TBD[128]|    Length     |  Source Plen  | Source Prefix...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
```

Fields:

Type   Set to TBD[128] to indicate a Source Prefix sub-TLV.

Length   The length of the body, exclusive of the Type and Length fields.

Source Plen The length of the advertised source prefix. This MUST NOT be 0.

Source Prefix The source prefix being advertised. This field's size is (Source Plen)/8 rounded upwards.

The source prefix encoding (AE) is the same as the Prefix's. It is defined by the AE field of the corresponding TLV.

Note that this sub-TLV is a Mandatory sub-TLV. The whole TLV MUST be ignored if that sub-TLV is not recognized. Otherwise, routing loops may occur (see Section 6.1).

7.2. Source-specific Update

The source-specific Update is an Update TLV with a Source Prefix sub-TLV. It advertises or retracts source-specific routes in the same manner than routes with non-source-specific Updates (see [BABEL]). A wildcard retraction (Update with AE equals to 0) MUST NOT carry a Source Prefix sub-TLV.

Contrary to the destination prefix, this extension does not compress the source prefix attached to Updates. However, as defined in [BABEL] (Section 4.5), the compression is allowed for the destination prefix of source-specific routes. Legacy implementation will correctly update their parser state while ignoring the whole TLV afterwards.

7.3. Source-specific (Route) Request

A source-specific Route Request is a Route Request TLV with a Source Prefix sub-TLV. It prompts the receiver to send an update for a given pair of destination and source prefixes. A wildcard request (Route Request with AE equals to 0) MUST NOT carry a Source Prefix sub-TLV.

7.4. Source-Specific Seqno Request

A source-specific Seqno Request is a Seqno Request TLV with a Source Prefix sub-TLV. It is just like a Seqno Request for a source-specific route. It uses the same mechanisms described in [BABEL].

8. IANA Considerations

IANA is requested to allocate TBD, a Babel sub-TLV type from the range reserved for mandatory sub-TLVs [value 128 suggested], and to add the following entry to the "Babel mandatory sub-TLV Types" registry:

Type	Name	Reference
TBD[128]	Source Prefix	(this document)

9. Security considerations

The extension defined in this document adds a new sub-TLV to three TLVs already present in the original Babel protocol. It does not by itself change the security properties of the protocol.

10. References

10.1. Normative References

[BABEL] Chroboczek, J., "The Babel Routing Protocol", Internet Draft draft-ietf-babel-rfc6126bis-02, May 2017.

[BCP84] Baker, F. and P. Savola, "Ingress Filtering for Multihomed Networks", BCP 84, RFC 3704, March 2004.

[IETF-SSR] Lamparter, D. and A. Smirnov, "Destination/Source Routing", Internet Draft draft-ietf-rtgwg-dst-src-routing, May 2017.

[RFC6126] Chroboczek, J., "The Babel Routing Protocol (Experimental)", RFC 6126, February 2011.

10.2. Informative References

[SS-ROUTING] Boutier, M. and J. Chroboczek, "Source-Specific Routing", August 2014.

In Proc. IFIP Networking 2015. A slightly earlier version is available online from http://arxiv.org/pdf/1403.0445.

Authors' Addresses

Matthieu Boutier
IRIF, University of Paris-Diderot
Case 7014
75205 Paris Cedex 13
France

Email: boutier@irif.fr

Juliusz Chroboczek
IRIF, University of Paris-Diderot
Case 7014
75205 Paris Cedex 13
France

Email: jch@irif.fr