Network Working Group
Internet-Draft
Updates: 6126 (if approved)
Intended status: Experimental
Expires: May 24, 2015

M. Boutier
J. Chroboczek
PPS, University of Paris-Diderot

November 20, 2014

Source-Specific Routing in Babel
draft-boutier-babel-source-specific-00

Abstract

This document describes extensions to the Babel routing protocol to support source-specific routing.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on May 24, 2015.

Copyright Notice

Table of Contents

1. Introduction and background

Source-specific routing is an extension to traditional next-hop where packets are routed according to both their destination and their source address. This document describes extensions to the Babel routing protocol [BABEL] to support source-specific routing.

Background information about source-specific routing is provided in [SS-ROUTING].

2. Data Structures

This extension adds some data to the data structures maintained by a Babel node.

2.1. The Source Table

Every Babel node maintains a source table, as described in [BABEL], Section 3.2.4. A source-specific Babel node extends this table with the following field:

- the source prefix (sprefix, splen) specifying the source of packets to which this entry applies.

If splen is 0, then this is a non-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. Notwithstanding the accidental similarity in their names, these are two very different concepts, and should not be confused.

2.2. The Route Table

Every Babel node maintains a route table, as described in [BABEL], Section 3.2.5. With this extension, this table is indexed by the 5-tuple (prefix, plen, source prefix, source plen, router-id) obtained from the associated source table entry.

2.3. The Table of Pending Requests

Every Babel node maintains a table of pending requests, as described in [BABEL], Section 3.2.6. A source-specific Babel node extends this table with the following entry:

- the source prefix being requested.

3. 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 prefix: two routing table entries might match a given packet without one necessarily being more specific than the other. Consider for example the following fragment of a routing table:

(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 packets with a 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.

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 layers that perform 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] for such an algorithm).
- If the lower layers cannot hold source-specific routes, then a Babel implementation SHOULD silently ignore any source-specific routes and MUST NOT reannounce them to other nodes.

4. Protocol Operation

This extension does not fundamentally change the operation of the Babel protocol.

4.1. Sending updates

This extension introduces a new kind of update, the source-specific update. Whenever a source-specific Babel node needs to send an update, it checks whether the update is for a source-specific route (a route with a source prefix of non-zero length); if that is the case, it sends a source-specific update (Section 5.1), and otherwise it sends a non-specific update ([BABEL], Section 4.4.9).

Every Babel node maintains a source table, which it updates whenever it sends an Update ([BABEL], Section 3.7.3). A source-specific Babel node extends this procedure by updating the source table not only when it sends an update, but also when it sends a source-specific update.

4.2. Requests

This extension duplicates Babel's two request types: there are now two kinds of route requests (source-specific and unspecific), and, similarly, two kinds of seqno requests.

This extension does not modify Babel's strategy for sending requests. Whenever a Babel node needs to send a request, it checks whether the request is for a source-specific route; if it is, it should send one of the request types defined in this document; if it is not, then it should send one of the request types defined in the original Babel specification.

4.2.1. Wildcard requests

The Babel protocol provides the ability to request a full routing table dump by sending a "wildcard request", a route request with the AE field set to 0. This extension does not modify the semantics of wildcard requests: a wildcard request prompts a dump of non-specific routes only, and a Babel node SHOULD NOT send any source-specific updates in reply to a wildcard request.

A different request is used for obtaining a dump of the source-specific routes in a node's routing table. A "source-specific wildcard request" is a source-specific request (Section 5.2) whose AE field is 0; it requests a dump of the receiving nodes source-specific routes only (routes with a source prefix length of 1 or more). A node SHOULD NOT send any non-specific updates in reply to a source-

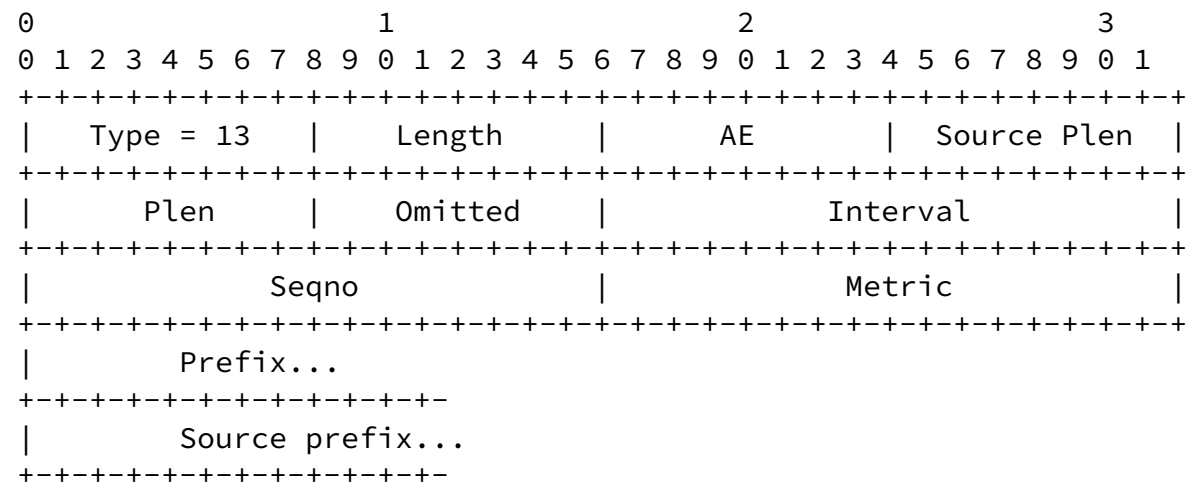
specific wildcard request.

In consequence, a node requiring a full routing table dump SHOULD send both a non-specific wildcard request and a source-specific wildcard request.

5. Protocol Encoding

This extension defines three new TLV types that are used by Source-Specific Babel nodes and silently ignored by ordinary Babel nodes, in accordance with [BABEL-EXT].

5.1. Source-Specific Update



Fields:

Type Set to 13 to indicate a source-specific update TLV.

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

AE The encoding of the prefix field.

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

Plen The length of the advertised destination prefix.

Omitted The number of octets that have been omitted at the beginning of the advertised prefix and that should be taken from a preceding Update TLV with the flag with value 80 hexadecimal set.

Interval An upper bound, expressed in centiseconds, on the time after which the sending node will send a new update for this prefix. This MUST NOT be 0 and SHOULD NOT be less than 10. The receiving node will use this value to compute a hold time for this routing table entry. The value FFFF hexadecimal (infinity) expresses that this announcement will not be repeated unless a request is received.

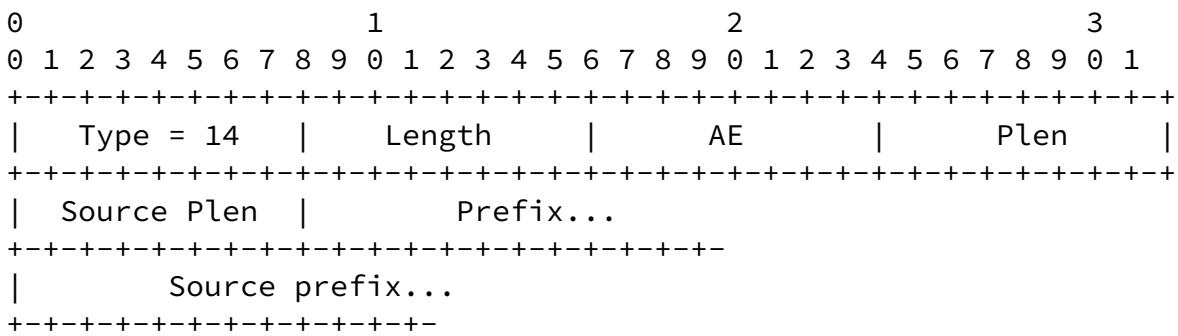
Seqno The originator's sequence number for this update.

Metric The sender's metric for this route. The value FFFF hexadecimal (infinity) means that this is a route retraction.

Prefix The destination prefix being advertised. This field's size is (Plen/8 - Omitted) rounded upwards.

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

5.2. Source-Specific Request



Fields:

Type Set to 14 to indicate a Source-Specific Route Request TLV.

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

AE The encoding of the prefix field. The value 0 (wildcard) requests a full dump of all the source-specific routes of the receving node (i.e. those with non-zero source prefix length).

Plen The length of the requested destination prefix.

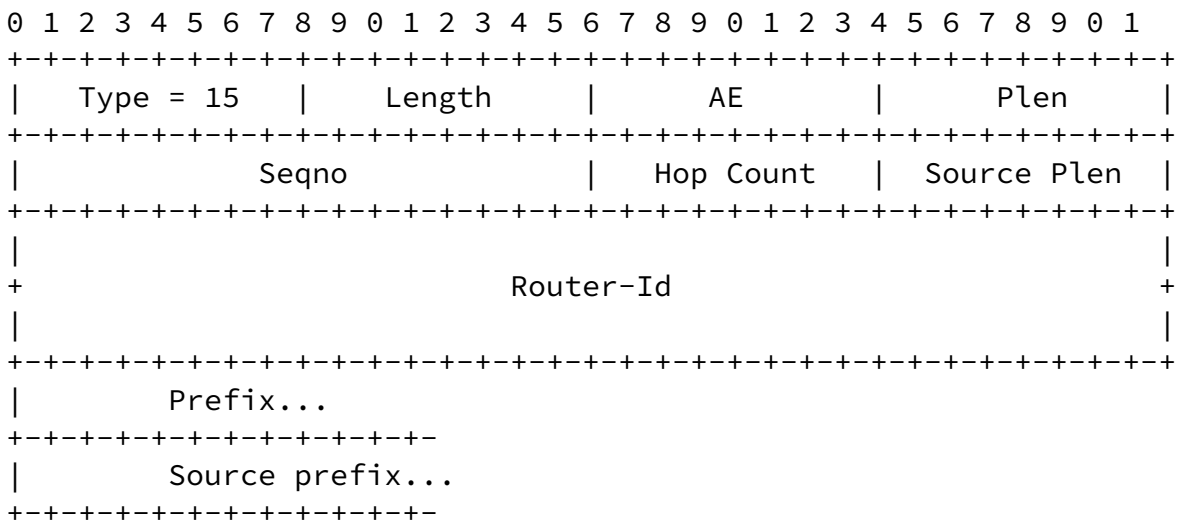
Source Plen The length of the requested source prefix. This MUST NOT be 0, except for AE 0.

Prefix The destination prefix being requested. This field's size is Plen/8 rounded upwards.

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

5.3. Source-Specific Seqno Request

0 1 2 3



Fields:

Type Set to 15 to indicate a Source-Specific Seqno Request TLV.

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

AE The encoding of the prefix field. This MUST NOT be 0.

Plen The length of the requested destination prefix.

Seqno The sequence number that is being requested.

Hop Count The maximum number of times that this TLV may be forwarded, plus 1. This MUST NOT be 0.

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

Router-Id The Router-ID associated to this route.

Prefix The destination prefix being requested. This field's size is Plen/8 rounded upwards.

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

A Source-Specific Seqno Request TLV prompts the receiving node to send an Update for the route specified by the AE, Plen, Prefix, Source Plen and Source Prefix fields, with either a router-id different from what is specified by the Router-Id field, or a Seqno no less than what is specified by the Seqno field. If this request

cannot be satisfied locally, then it is forwarded according to the rules set out in Section 3.8.1.2 of [BABEL].

Just like an ordinary Seqno Request, a Source-Specific Seqno Request MAY be sent to a multicast address but MUST NOT be forwarded to a multicast address and MUST NOT be forwarded to more than one neighbour. A Source-Specific Seqno Request MUST NOT be forwarded if its Hop Count field is 1.

6. IANA Considerations

This document defines three new Babel TLV types:

13 - Source-Specific Update

14 - Source-Specific Request

15 - Source-Specific Seqno Request

7. Security considerations

By itself, the Babel routing protocol is not a secure protocol; this extension doesn't change that fact. Babel can be made secure by using a suitable cryptographic layer, such as the one described in [RFC7298].

8. References

[BABEL] Chroboczek, J., "The Babel Routing Protocol", RFC 6126, February 2011.

[BABEL-EXT] Chroboczek, J., "Extension Mechanism for the Babel Routing Protocol", Internet Draft draft-chroboczek-babel-extension-mechanism-00, June 2013.

[RFC7298] Ovsienko, D., "Babel Hashed Message Authentication Code (HMAC) Cryptographic Authentication", RFC 7298, July 2014.

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

Authors' Addresses

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

Email: boutier@pps.univ-paris-diderot.fr

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

Email: jch@pps.univ-paris-diderot.fr