
Stream: Internet Engineering Task Force (IETF)
RFC: [8965](#)
Category: Informational
Published: January 2021
ISSN: 2070-1721
Author: J. Chroboczek
IRIF, University of Paris-Diderot

RFC 8965

Applicability of the Babel Routing Protocol

Abstract

Babel is a routing protocol based on the distance-vector algorithm augmented with mechanisms for loop avoidance and starvation avoidance. This document describes a number of niches where Babel has been found to be useful and that are arguably not adequately served by more mature protocols.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Not all documents approved by the IESG are candidates for any level of Internet Standard; see Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at <https://www.rfc-editor.org/info/rfc8965>.

Copyright Notice

Copyright (c) 2021 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction and Background
 - 1.1. Technical Overview of the Babel Protocol
2. Properties of the Babel Protocol
 - 2.1. Simplicity and Implementability
 - 2.2. Robustness
 - 2.3. Extensibility
 - 2.4. Limitations
3. Successful Deployments of Babel
 - 3.1. Heterogeneous Networks
 - 3.2. Large-Scale Overlay Networks
 - 3.3. Pure Mesh Networks
 - 3.4. Small Unmanaged Networks
4. Security Considerations
5. References
 - 5.1. Normative References
 - 5.2. Informative References

Acknowledgments

Author's Address

1. Introduction and Background

Babel [RFC8966] is a routing protocol based on the familiar distance-vector algorithm (sometimes known as distributed Bellman-Ford) augmented with mechanisms for loop avoidance (there is no "counting to infinity") and starvation avoidance. This document describes a number of niches where Babel is useful and that are arguably not adequately served by more mature protocols such as OSPF [RFC5340] and IS-IS [RFC1195].

1.1. Technical Overview of the Babel Protocol

At its core, Babel is a distance-vector protocol based on the distributed Bellman-Ford algorithm, similar in principle to RIP [RFC2453] but with two important extensions: provisions for sensing of neighbour reachability, bidirectional reachability, and link quality, and support for multiple address families (e.g., IPv6 and IPv4) in a single protocol instance.

Algorithms of this class are simple to understand and simple to implement, but unfortunately they do not work very well -- they suffer from "counting to infinity", a case of pathologically slow convergence in some topologies after a link failure. Babel uses a mechanism pioneered by the Enhanced Interior Gateway Routing Protocol (EIGRP) [DUAL] [RFC7868], known as "feasibility", which avoids routing loops and therefore makes counting to infinity impossible.

Feasibility is a conservative mechanism, one that not only avoids all looping routes but also rejects some loop-free routes. Thus, it can lead to a situation known as "starvation", where a router rejects all routes to a given destination, even those that are loop-free. In order to recover from starvation, Babel uses a mechanism pioneered by the Destination-Sequenced Distance-Vector Routing Protocol (DSDV) [DSDV] and known as "sequenced routes". In Babel, this mechanism is generalised to deal with prefixes of arbitrary length and routes announced at multiple points in a single routing domain (DSDV was a pure mesh protocol, and only carried host routes).

In DSDV, the sequenced routes algorithm is slow to react to a starvation episode. In Babel, starvation recovery is accelerated by using explicit requests (known as "seqno requests" in the protocol) that signal a starvation episode and cause a new sequenced route to be propagated in a timely manner. In the absence of packet loss, this mechanism is provably complete and clears the starvation in time proportional to the diameter of the network, at the cost of some additional signalling traffic.

2. Properties of the Babel Protocol

This section describes the properties of the Babel protocol as well as its known limitations.

2.1. Simplicity and Implementability

Babel is a conceptually simple protocol. It consists of a familiar algorithm (distributed Bellman-Ford) augmented with three simple and well-defined mechanisms (feasibility, sequenced routes, and explicit requests). Given a sufficiently friendly audience, the principles behind Babel can be explained in 15 minutes, and a full description of the protocol can be done in 52 minutes (one microcentury).

An important consequence is that Babel is easy to implement. At the time of writing, there exist four independent, interoperable implementations, including one that was reportedly written and debugged in just two nights.

2.2. Robustness

The fairly strong properties of the Babel protocol (convergence, loop avoidance, and starvation avoidance) rely on some reasonably weak properties of the network and the metric being used. The most significant are:

causality: the "happens-before" relation is acyclic (intuitively, a control message is not received before it has been sent);

strict monotonicity of the metric: for any metric M and link cost C , $M < C + M$ (intuitively, this implies that cycles have a strictly positive metric);

left-distributivity of the metric: for any metrics M and M' and cost C , if $M \leq M'$, then $C + M \leq C + M'$ (intuitively, this implies that a good choice made by a neighbour B of a node A is also a good choice for A).

See [\[METAROUTING\]](#) for more information about these properties and their consequences.

In particular, Babel does not assume a reliable transport, it does not assume ordered delivery, it does not assume that communication is transitive, and it does not require that the metric be discrete (continuous metrics are possible, for example, reflecting packet loss rates). This is in contrast to link-state routing protocols such as OSPF [\[RFC5340\]](#) or IS-IS [\[RFC1195\]](#), which incorporate a reliable flooding algorithm and make stronger requirements on the underlying network and metric.

These weak requirements make Babel a robust protocol:

robust with respect to unusual networks: an unusual network (non-transitive links, unstable link costs, etc.) is likely not to violate the assumptions of the protocol;

robust with respect to novel metrics: an unusual metric (continuous, constantly fluctuating, etc.) is likely not to violate the assumptions of the protocol.

[Section 3](#) gives examples of successful deployments of Babel that illustrate these properties.

These robustness properties have important consequences for the applicability of the protocol: Babel works (more or less efficiently) in a range of circumstances where traditional routing protocols don't work well (or at all).

2.3. Extensibility

Babel's packet format has a number of features that make the protocol extensible (see [Appendix D](#) of [\[RFC8966\]](#)), and a number of extensions have been designed to make Babel work better in situations that were not envisioned when the protocol was initially designed. The ease of extensibility is not an accident, but a consequence of the design of the protocol: it is reasonably easy to check whether a given extension violates the assumptions on which Babel relies.

All of the extensions designed to date interoperate with the base protocol and with each other. This, again, is a consequence of the protocol design: in order to check that two extensions to the Babel protocol are interoperable, it is enough to verify that the interaction of the two does not violate the base protocol's assumptions.

Notable extensions deployed to date include:

- source-specific routing (also known as Source-Address Dependent Routing, SADR) [[BABEL-SS](#)] allows forwarding to take a packet's source address into account, thus enabling a cheap form of multihoming [[SS-ROUTING](#)];
- RTT-based routing [[BABEL-RTT](#)] minimises link delay, which is useful in overlay network (where both hop count and packet loss are poor metrics).

Some other extensions have been designed but have not seen deployment in production (and their usefulness is yet to be demonstrated):

- frequency-aware routing [[BABEL-Z](#)] aims to minimise radio interference in wireless networks;
- ToS-aware routing [[BABEL-TOS](#)] allows routing to take a packet's Type of Service (ToS) marking into account for selected routes without incurring the full cost of a multi-topology routing protocol.

2.4. Limitations

Babel has some undesirable properties that make it suboptimal or even unusable in some deployments.

2.4.1. Periodic Updates

The main mechanisms used by Babel to reconverge after a topology change are reactive: triggered updates, triggered retractions and explicit requests. However, Babel relies on periodic updates to clear pathologies after a mobility event or in the presence of heavy packet loss. The use of periodic updates makes Babel unsuitable in at least two kinds of environments:

large, stable networks: since Babel sends periodic updates even in the absence of topology changes, in well-managed, large, stable networks the amount of control traffic will be reduced by using a protocol that uses a reliable transport (such as OSPF, IS-IS, or EIGRP);

low-power networks: the periodic updates use up battery power even when there are no topology changes and no user traffic, which makes Babel wasteful in low-power networks.

2.4.2. Full Routing Table

While there exist techniques that allow a Babel speaker to function with a partial routing table (e.g., by learning just a default route or, more generally, performing route aggregation), Babel is designed around the assumption that every router has a full routing table. In networks where some nodes are too constrained to hold a full routing table, it might be preferable to use a protocol that was designed from the outset to work with a partial routing table (such as the Ad

hoc On-Demand Distance Vector (AODV) routing protocol [[RFC3561](#)], the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) [[RFC6550](#)], or the Lightweight On-demand Ad hoc Distance-vector Routing Protocol - Next Generation (LOADng) [[LOADng](#)].

2.4.3. Slow Aggregation

Babel's loop-avoidance mechanism relies on making a route unreachable after a retraction until all neighbours have been guaranteed to have acted upon the retraction, even in the presence of packet loss. Unless the second algorithm described in [Section 3.5.5](#) of [[RFC8966](#)] is implemented, this entails that a node is unreachable for a few minutes after the most specific route to it has been retracted. This delay makes Babel slow to recover from a topology change in networks that perform automatic route aggregation.

3. Successful Deployments of Babel

This section gives a few examples of environments where Babel has been successfully deployed.

3.1. Heterogeneous Networks

Babel is able to deal with both classical, prefix-based ("Internet-style") routing and flat ("mesh-style") routing over non-transitive link technologies. Just like traditional distance-vector protocols, Babel is able to carry prefixes of arbitrary length, to suppress redundant announcements by applying the split-horizon optimisation where applicable, and can be configured to filter out redundant announcements (manual aggregation). Just like specialised mesh protocols, Babel doesn't by default assume that links are transitive or symmetric, can dynamically compute metrics based on an estimation of link quality, and carries large numbers of host routes efficiently by omitting common prefixes.

Because of these properties, Babel has seen a number of successful deployments in medium-sized heterogeneous networks, networks that combine a wired, aggregated backbone with meshy wireless bits at the edges.

Efficient operation in heterogeneous networks requires the implementation to distinguish between wired and wireless links, and to perform link quality estimation on wireless links.

3.2. Large-Scale Overlay Networks

The algorithms used by Babel (loop avoidance, hysteresis, delayed updates) allow it to remain stable in the presence of unstable metrics, even in the presence of a feedback loop. For this reason, it has been successfully deployed in large-scale overlay networks, built out of thousands of tunnels spanning continents, where it is used with a metric computed from links' latencies.

This particular application depends on the extension for RTT-sensitive routing [[DELAY-BASED](#)].

3.3. Pure Mesh Networks

While Babel is a general-purpose routing protocol, it has been shown to be competitive with dedicated routing protocols for wireless mesh networks [REAL-WORLD] [BRIDGING-LAYERS]. Although this particular niche is already served by a number of mature protocols, notably the Optimized Link State Routing Protocol with Expected Transmission Count (OLSR-ETX) and OLSRv2 (OLSR Version 2) [RFC7181] (equipped e.g., with the Directional Airtime (DAT) metric [RFC7779]), Babel has seen a moderate amount of successful deployment in pure mesh networks.

3.4. Small Unmanaged Networks

Because of its small size and simple configuration, Babel has been deployed in small, unmanaged networks (e.g., home and small office networks), where it serves as a more efficient replacement for RIP [RFC2453], over which it has two significant advantages: the ability to route multiple address families (IPv6 and IPv4) in a single protocol instance and good support for using wireless links for transit.

4. Security Considerations

As is the case in all distance-vector routing protocols, a Babel speaker receives reachability information from its neighbours, which by default is trusted by all nodes in the routing domain.

At the time of writing, the Babel protocol is usually run over a network that is secured either at the physical layer (e.g., physically protecting Ethernet sockets) or at the link layer (using a protocol such as Wi-Fi Protected Access 2 (WPA2)). If Babel is being run over an unprotected network, then the routing traffic needs to be protected using a sufficiently strong cryptographic mechanism.

At the time of writing, two such mechanisms have been defined. Message Authentication Code (MAC) authentication for Babel (Babel-MAC) [RFC8967] is a simple and easy to implement mechanism that only guarantees authenticity, integrity, and replay protection of the routing traffic and only supports symmetric keying with a small number of keys (typically just one or two). Babel-DTLS [RFC8968] is a more complex mechanism that requires some minor changes to be made to a typical Babel implementation and depends on a DTLS stack being available, but inherits all of the features of DTLS, notably confidentiality, optional replay protection, and the ability to use asymmetric keys.

Due to its simplicity, Babel-MAC should be the preferred security mechanism in most deployments, with Babel-DTLS available for networks that require its additional features.

In addition to the above, the information that a mobile Babel node announces to the whole routing domain is often sufficient to determine a mobile node's physical location with reasonable precision. This might make Babel unapplicable in scenarios where a node's location is considered confidential.

5. References

5.1. Normative References

- [RFC8966] Chroboczek, J. and D. Schinazi, "The Babel Routing Protocol", RFC 8966, DOI 10.17487/RFC8966, January 2021, <<https://www.rfc-editor.org/info/rfc8966>>.

5.2. Informative References

- [BABEL-RTT] Jonglez, B. and J. Chroboczek, "Delay-based Metric Extension for the Babel Routing Protocol", Work in Progress, Internet-Draft, draft-jonglez-babel-rtt-extension-02, 11 March 2019, <<https://tools.ietf.org/html/draft-jonglez-babel-rtt-extension-02>>.
- [BABEL-SS] Boutier, M. and J. Chroboczek, "Source-Specific Routing in Babel", Work in Progress, Internet-Draft, draft-ietf-babel-source-specific-07, 28 October 2020, <<https://tools.ietf.org/html/draft-ietf-babel-source-specific-07>>.
- [BABEL-TOS] Chouasne, G. and J. Chroboczek, "TOS-Specific Routing in Babel", Work in Progress, Internet-Draft, draft-chouasne-babel-tos-specific-00, 3 July 2017, <<https://tools.ietf.org/html/draft-chouasne-babel-tos-specific-00>>.
- [BABEL-Z] Chroboczek, J., "Diversity Routing for the Babel Routing Protocol", Work in Progress, Internet-Draft, draft-chroboczek-babel-diversity-routing-01, 15 February 2016, <<https://tools.ietf.org/html/draft-chroboczek-babel-diversity-routing-01>>.
- [BRIDGING-LAYERS] Murray, D., Dixon, M., and T. Koziniec, "An Experimental Comparison of Routing Protocols in Multi Hop Ad Hoc Networks", In Proceedings of ATNAC, DOI 10.1109/ATNAC.2010.5680190, October 2010, <<https://doi.org/10.1109/ATNAC.2010.5680190>>.
- [DELAY-BASED] Jonglez, B., Boutier, M., and J. Chroboczek, "A delay-based routing metric", March 2014, <<http://arxiv.org/abs/1403.3488>>.
- [DSDV] Perkins, C. and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers", ACM SIGCOMM '94: Proceedings of the Conference on Communications Architectures, Protocols and Applications, pp. 234-244, DOI 10.1145/190314.190336, October 1994, <<https://doi.org/10.1145/190314.190336>>.
- [DUAL] Garcia-Luna-Aceves, J. J., "Loop-Free Routing Using Diffusing Computations", IEEE/ACM Transactions on Networking, Volume 1, Issue 1, DOI 10.1109/90.222913, February 1993, <<https://doi.org/10.1109/90.222913>>.

-
- [LOADng]** Clausen, T. H., Verdiere, A. C. D., Yi, J., Niktash, A., Igarashi, Y., Satoh, H., Herberg, U., Lavenu, C., Lys, T., and J. Dean, "The Lightweight On-demand Ad hoc Distance-vector Routing Protocol - Next Generation (LOADng)", Work in Progress, Internet-Draft, draft-clausen-lln-loadng-15, 4 July 2016, <<https://tools.ietf.org/html/draft-clausen-lln-loadng-15>>.
- [METAROUTING]** Griffin, T. G. and J. L. Sobrinho, "Metarouting", ACM SIGCOMM Computer Communication Review, Volume 35, Issue 4, DOI 10.1145/1090191.1080094, August 2005, <<https://doi.org/10.1145/1090191.1080094>>.
- [REAL-WORLD]** Abolhasan, M., Hagelstein, B., and J. C.-P. Wang, "Real-world performance of current proactive multi-hop mesh protocols", 15th Asia-Pacific Conference on Communications, DOI 10.1109/APCC.2009.5375690, October 2009, <<https://doi.org/10.1109/APCC.2009.5375690>>.
- [RFC1195]** Callon, R., "Use of OSI IS-IS for routing in TCP/IP and dual environments", RFC 1195, DOI 10.17487/RFC1195, December 1990, <<https://www.rfc-editor.org/info/rfc1195>>.
- [RFC2453]** Malkin, G., "RIP Version 2", STD 56, RFC 2453, DOI 10.17487/RFC2453, November 1998, <<https://www.rfc-editor.org/info/rfc2453>>.
- [RFC3561]** Perkins, C., Belding-Royer, E., and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing", RFC 3561, DOI 10.17487/RFC3561, July 2003, <<https://www.rfc-editor.org/info/rfc3561>>.
- [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>>.
- [RFC6550]** Winter, T., Ed., Thubert, P., Ed., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, JP., and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks", RFC 6550, DOI 10.17487/RFC6550, March 2012, <<https://www.rfc-editor.org/info/rfc6550>>.
- [RFC7181]** Clausen, T., Dearlove, C., Jacquet, P., and U. Herberg, "The Optimized Link State Routing Protocol Version 2", RFC 7181, DOI 10.17487/RFC7181, April 2014, <<https://www.rfc-editor.org/info/rfc7181>>.
- [RFC7779]** Rogge, H. and E. Baccelli, "Directional Airtime Metric Based on Packet Sequence Numbers for Optimized Link State Routing Version 2 (OLSRv2)", RFC 7779, DOI 10.17487/RFC7779, April 2016, <<https://www.rfc-editor.org/info/rfc7779>>.
- [RFC7868]** Savage, D., Ng, J., Moore, S., Slice, D., Paluch, P., and R. White, "Cisco's Enhanced Interior Gateway Routing Protocol (EIGRP)", RFC 7868, DOI 10.17487/RFC7868, May 2016, <<https://www.rfc-editor.org/info/rfc7868>>.
- [RFC8967]** Dô, C., Kolodziejak, W., and J. Chroboczek, "MAC Authentication for the Babel Routing Protocol", RFC 8967, DOI 10.17487/RFC8967, January 2021, <<https://www.rfc-editor.org/info/rfc8967>>.
-

- [RFC8968]** Décimo, A., Schinazi, D., and J. Chroboczek, "Babel Routing Protocol over Datagram Transport Layer Security", RFC 8968, DOI 10.17487/RFC8968, January 2021, <<https://www.rfc-editor.org/info/rfc8968>>.
- [SS-ROUTING]** Boutier, M. and J. Chroboczek, "Source-specific routing", In Proceedings of the IFIP Networking Conference, DOI 10.1109/IFIPNetworking.2015.7145305, May 2015, <<http://arxiv.org/pdf/1403.0445>>.

Acknowledgments

The author is indebted to Jean-Paul Smetz and Alexander Vainshtein for their input to this document.

Author's Address

Juliusz Chroboczek
IRIF, University of Paris-Diderot
Case 7014
75205 Paris CEDEX 13
France
Email: jch@irif.fr