Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: June 19, 2016

B. Decraene
Orange
S. Litkowski
Orange Business Service
H. Gredler
Private Contributor
A. Lindem
P. Francois
Cisco Systems
December 17, 2015

SPF Back-off algorithm for link state IGPs
draft-ietf-rtgwg-backoff-algo-02

Abstract

This document defines a standard algorithm to back-off link-state IGP SPF computations.

Having one standard algorithm improves interoperability by reducing the probability and/or duration of transient forwarding loops during the IGP convergence when the IGP reacts to multiple proximate IGP events.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

Status of This Memo

Table of Contents

1. Introduction

Link state IGPs, such as IS-IS [ISO10589-Second-Edition] and OSPF [RFC2328], perform distributed route computation on all routers of the area/level. In order to have consistent routing tables across the network, such distributed computation requires that all routers have the same version of the network topology (Link State DataBase (LSDB)) and perform their computation at the same time.

In general, when the network is stable, there is a desire to compute the new SPF as soon as the failure is detected in order to quickly route around the failure. However, when the network is experiencing multiple proximate failures over a short period of time, there is a conflicting desire to limit the frequency of SPF computations. Indeed, this allows a reduction in control plane resources used by IGPs and all protocols/subsystem reacting on the attendant route

change, such as LDP, RSVP-TE, BGP, Fast ReRoute computations, FIB updates... This will reduce the churn on routers and in the network and, in particular, reduce the side effects such as micro-loops that ensue during IGP convergence.

To allow for this, IGPs implement a SPF back-off algorithm. Different implementations choose different algorithms. Hence, in a multi-vendor network, it's not possible to ensure that all routers trigger their SPF computation after the same delay. This situation increases the average differential delay between routers completing their SPF computation. It also increases the probability that different routers compute their FIBs based on a different LSDB versions. Both factors increase the probability and/or duration of micro-loops.

To allow multi-vendor networks to have all routers delay their SPF computations for the same duration, this document specifies a standard algorithm. Optionally, implementations may offer alternative algorithms.

2. High level goals

The high level goals of this algorithm are the following:

- Very fast convergence for a single event (e.g., link failure).
- Slightly paced fast convergence for multiple proximate IGP events while IGP stability is considered acceptable.
- Delayed convergence when the IGP stability is problematic. This will allow the IGP and related processes to conserve resources during the period of instability.
- Always try to avoid different SPF_DELAY timers values across different routers in the area/level. Even though not all routers will receive IGP messages at the same time (due to differences both in the distance from the originator of the IGP event and in flooding implementations).

3. Definitions and parameters

IGP events: An IGP LSDB change requiring a new routing table computation. Examples are a topology change, a prefix change, a metric change on link or prefix...

Routing table computation: computation of the routing table, by the IGP, using the IGP LSDB. No distinction is made between the type of computation performed. e.g., full SPF, incremental SPF, Partial Route

Computation (PRC). The type of computation is a local consideration. This document may indifferently use the terms routing table computation or SPF computation.

The SPF_DELAY is the delay introduced between the IGP event and the start of the routing table computation. It can take the following values:

INITIAL_WAIT: a very small delay to quickly handle link failure, e.g., 0 milliseconds.

FAST_WAIT: a small delay to have a fast convergence in case of single component failure (node, SRLG..), e.g., 50-100 milliseconds. Note: we want to be fast, but as this failure results in multiple IGP events, being too fast increases the probability to receive additional network events immediately after the SPF computation.

LONG_WAIT: a long delay when the IGP is unstable, e.g., 2 seconds. Note: Allow the IGP network to stabilize.

The TIME_TO_LEARN timer is the maximum duration typically needed to learn all the IGP events related to a single component failure (e.g., router failure, SRLG failure), e.g., 1 second. It's mostly dependent on variation of failure detection times between all routers that are adjacent to the failure. Additionally, it may depend on the different flooding implementations for routers in the network.

The HOLD_DOWN timer is the time needed with no received IGP events before considering the IGP to be stable again, allowing the SPF_DELAY to be restored to INITIAL_WAIT. e.g., 3 seconds.

4. Principles of SPF delay algorithm

For this first IGP event, we assume that there has been a single simple change in the network which can be taken into account using a single routing computation (e.g., link failure, prefix (metric) change) and we optimize for very fast convergence, delaying the routing computation by INITIAL_WAIT. Under this assumption, there is no benefit in delaying the routing computation. In a typical network, this is the most common type of IGP event. Hence, it makes sense to optimize this case.

If subsequent IGP events are received in a short period of time (TIME_TO_LEARN), we then assume that a single component failed, but that this failure requires the knowledge of multiple IGP events in order for the IGP routing to converge. Under this assumption, we want fast convergence since this is a normal network situation. However, there is a benefit in waiting for all IGP events related to

this single component failure so that the IGP can compute the post-failure routing table in a single route computation. In this situation, we delay the routing computation by FAST_WAIT.

If IGP events are still received after TIME_TO_LEARN seconds from the initial IGP event, then the network is presumably experiencing multiple independent failures and while waiting for network stability, the computations are delayed for a longer time represented by LONG_WAIT. This SPF_delay is kept until no IGP events are received for HOLD_DOWN seconds.

Note: previous SPF delay algorithms used to count the number of SPF computations. However, as all routers may receive the IGP events at different times, we cannot assume that all routers will perform the same number of SPF computations or that they will schedule them at the same time. For example, assuming that the SPF delay is 50 ms, router R1 may receive 3 IGP events (E1, E2, E3) in those 50 ms and hence will perform a single routing computation. While another router R2 may only receive 2 events (E1, E2) in those 50 ms and hence will schedule another routing computation when receiving E3. That's why this document defines a time (TIME_TO_LEARN) from the initial event detection/reception as opposed to defining the number of SPF computations to determine when the IGP is unstable.

5. Specification of the SPF delay algorithm

When no IGP events have occurred during the HOLD_DOWN interval:

- The IGP is set to the QUIET state.

When the IGP is in the QUIET state and an IGP event is received:

- The time of this first IGP event is stored in FIRST_EVENT_TIME.
- The next routing table computation is scheduled at: this IGP event received time + INITIAL_WAIT.
- The IGP is set to the FAST_WAIT state.

When the IGP is in the FAST_WAIT state and an IGP event is received:

- If more than the TIME_TO_LEARN interval has passed since FIRST_EVENT_TIME, then the IGP is set to the HOLD_DOWN state.
- If a routing table computation is not already scheduled, one is scheduled at: this IGP event received time + FAST_WAIT.

When the IGP is in the HOLD_DOWN state and an IGP event is received:

- o If a routing table computation is not already scheduled, one is scheduled at: this IGP event received time + LONG_WAIT.

6. Parameters

All the parameters MUST be configurable. All the delays (INITIAL_WAIT, FAST_WAIT, LONG_WAIT, TIME_TO_LEARN, HOLD_DOWN) SHOULD be configurable at the millisecond granularity. They MUST be configurable at least at the tenth of second granularity. The configurable range for all the parameters SHOULD be at least from 0 milliseconds to 60 seconds.

This document does not propose default values for the parameters because these values are expected to be context dependent. Implementations are free to propose their own default values.

When setting the (default) values, one SHOULD consider the customer's or their applications' requirements, the computational power of the routers, the size of the network, and, in particular, the number of IP prefixes advertised in the IGP, the frequency and number of IGP events, the number of protocols reactions/computations triggered by IGP SPF (e.g., BGP, PCEP, Traffic Engineering CSPF, Fast ReRoute computations).

Note that some or all of these factors may change over the life of the network. In case of doubt, it's RECOMMENDED to play it safe and start with safe, i.e., longer timers.

For the standard algorithm to be effective in mitigating micro-loops, it is RECOMMENDED that all routers in the IGP domain, or at least all the routers in the same area/level, have exactly the same configured values.

7. Impact on micro-loops

Micro-loops during IGP convergence are due to a non-synchronized or non-ordered update of the forwarding information tables (FIB) [RFC5715] [RFC6976] [I-D.ietf-rtgwg-spf-uloop-pb-statement]. FIBs are installed after multiple steps such as SPF wait time, SPF computation, FIB distribution, and FIB update. This document only addresses the first contribution. This standardized procedure reduces the probability and/or duration of micro-loops when the IGP experience multiple proximate events. It does not prevent all micro-loops. However, it is beneficial and its cost seems limited compared to full solutions such as [RFC5715] or [RFC6976].

8. IANA Considerations

No IANA actions required.

9. Security considerations

This algorithm presented in this document does not in any way compromise the security of the IGP. In fact, the HOLD_DOWN state may mitigate the effects of Denial-of-Service (DOS) attacks generating many IGP events.

10. Acknowledgements

We would like to acknowledge Les Ginsberg, Uma Chunduri, and Mike Shand for the discussions and comments related to this document.

11. References

11.1. 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, <http://www.rfc-editor.org/info/rfc2119>.

11.2. Informative References

[I-D.ietf-rtgwg-spf-uloop-pb-statement] Litkowski, S., Decraene, B., and M. Horneffer, "Link State protocols SPF trigger and delay algorithm impact on IGP micro-loops", draft-ietf-rtgwg-spf-uloop-pb-statement-02 (work in progress), December 2015.

[ISO10589-Second-Edition] International Organization for Standardization, "Intermediate system to Intermediate system intra-domain routeing information exchange protocol for use in conjunction with the protocol for providing the connectionless-mode Network Service (ISO 8473)", ISO/IEC 10589:2002, Second Edition, Nov 2002.

[RFC2328] Moy, J., "OSPF Version 2", STD 54, RFC 2328, DOI 10.17487/RFC2328, April 1998, <http://www.rfc-editor.org/info/rfc2328>.

[RFC5715] Shand, M. and S. Bryant, "A Framework for Loop-Free Convergence", RFC 5715, DOI 10.17487/RFC5715, January 2010, <http://www.rfc-editor.org/info/rfc5715>.

[RFC6976] Shand, M., Bryant, S., Previdi, S., Filsfils, C., Francois, P., and O. Bonaventure, "Framework for Loop-Free Convergence Using the Ordered Forwarding Information Base (oFIB) Approach", RFC 6976, DOI 10.17487/RFC6976, July 2013, <http://www.rfc-editor.org/info/rfc6976>.

Authors' Addresses

Bruno Decraene
Orange
38 rue du General Leclerc
Issy Moulineaux cedex 9 92794
France

Email: bruno.decraene@orange.com

Stephane Litkowski
Orange Business Service

Email: stephane.litkowski@orange.com

Hannes Gredler
Private Contributor

Email: hannes@gredler.at

Acee Lindem
Cisco Systems
301 Midenhall Way
Cary, NC 27513
USA

Email: acee@cisco.com

Pierre Francois
Cisco Systems

Email: pifranco@cisco.com