Network Working Group Manoj Leelanivas (Juniper Networks)
Internet Draft Yakov Rekhter(Juniper Networks)
Expiration Date: March 2002 Rahul Aggarwal (Redback Networks)

Graceful Restart Mechanism for LDP

draft-leelanivas-ldp-restart-01.txt

1. Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026, except that the right to produce derivative works is not granted.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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.''

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

2. Abstract

This document describes a mechanism that helps to minimize the negative effects on MPLS traffic caused by LSR's control plane restart, and specifically by the restart of its LDP component, on LSRs that are capable of preserving the MPLS forwarding component across the restart.

3. Summary for Sub-IP Area

3.1. Summary

This document describes a mechanism that helps to minimize the negative effects on MPLS traffic caused by LSR's control plane restart, and specifically by the restart of its LDP component, on LSRs that are capable of preserving the MPLS forwarding component across the restart.

3.2. Related documents

See the Reference Section

3.3. Where does it fit in the Picture of the Sub-IP Work

This work fits squarely in MPLS box.

3.4. Why is it Targeted at this WG

The LDP is a product of the MPLS WG. This document specifies procedures to minimize the negative effects caused by the restart of the control plane LDP module. Since the procedures described in this document are directly related to LDP, it would be logical to target this document at the MPLS WG.

3.5. Justification

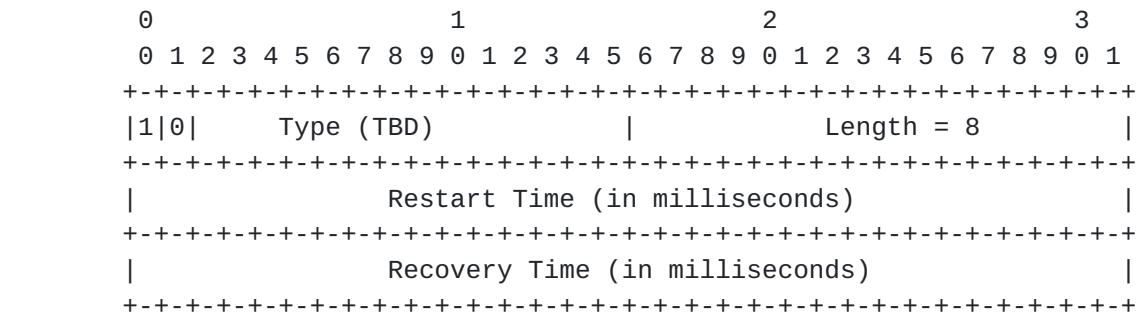
The WG should consider this document, as it allows to minimize the negative effects caused by the restart of the control plane LDP module.

4. Motivation

In the case where an LSR could preserve its MPLS forwarding state across restart of its control plane, and specifically its LDP component [LDP], it may be desirable not to perturb the LSPs going through that LSR (and specifically, the LSPs established by LDP). In this document, we describe a mechanism, termed "LDP Graceful Restart", that allows to accomplish this goal.

5. LDP Extension

An LSR indicates that it is capable of supporting LDP Graceful Restart, as defined in this document, by including the Graceful Restart TLV as an Optional Parameter in the LDP Initialization message.



The value field of the Graceful Restart TLV contains two components: Restart Time, and Recovery Time.

The Restart Time is the time (in milliseconds) that the sender of the TLV would like the receiver of that TLV to wait after the receiver detects the failure of LDP communication with the sender. While waiting, the receiver should retain the LDP and MPLS forwarding state for the (already established) LSPs that traverse a link between the sender and the receiver. The Restart Time should be long enough to allow the restart of the control plane of the sender of the TLV, and specifically its LDP component to bring it to the state where the sender could exchange LDP messages with its neighbors.

For a restarting LSR the Recovery Time carries the time (in milliseconds) the LSR is willing to retain its MPLS forwarding state that it preserved across the restart. The time is from the moment the LSR sends the Initialization message that carries the Graceful Restart TLV after restart. Setting this time to 0 indicates that the MPLS forwarding state wasn't preserved across the restart (or even if it was preserved, is no longer available).

For an (non restarting) LSR that re-established an LDP adjacency with a neighbor, this is the time (in milliseconds) that the LSR is willing to retain the label-FEC bindings that have been received from the neighbor prior to neighbor's restart. The time is from the moment the LSR sends the Initialization message that carries the Graceful Restart TLV.

The Recovery Time should be long enough to allow the neighboring LSR's to re-sync all the LSP's in a graceful manner, without creating congestion in the LDP control plane.

6. Operations

For the sake of brevity in the context of this document by "the control plane" we mean "the LDP component of the control plane".

An LSR that supports functionality described in this document should advertise this to its LDP neighbors by carrying the Graceful Restart TLV in the LDP Initialization message.

The procedures described in this document apply to downstream unsolicited label distribution. Extending these procedures to downstream on demand label distribution is for further study.

This document assumes that in addition to the MPLS forwarding state, an LSR can also preserve its IP forwarding state across the restart. Procedures for preserving IP forwarding state across the restart are defined in [OSPF-RESTART], [ISIS-RESTART], and [BGP-RESTART].

6.1. Procedures for the restarting LSR

For the sake of brevity in the context of this document by "MPLS forwarding state" we mean either <incoming label -> (outgoing label, next hop)> (non-ingress case), or <FEC->(outgoing label, next hop)> (ingress case) mapping.

After an LSR restarts its control plane, the LSR should check whether it was able to preserve its MPLS forwarding state from prior to the restart. If no, then the LSR must set the Recovery Time to 0 in the Graceful Restart TLV the LSR sends to its neighbors.

If the forwarding state has been preserved, then the LSR starts its internal timer, called MPLS Forwarding State Holding timer (the value of that timer should be configurable), and marks all the MPLS forwarding state entries as "stale". At the expiration of the timer, all the entries still marked as stale should be deleted. The value of

the Recovery Time advertised in the Graceful Restart TLV should be set to the (current) value of the timer at the point when the Initialization message carrying the Graceful Restart TLV is sent.

We say that an LSR is in the process of restarting when the MPLS Forwarding State Holding timer is not expired. Once the timer expires, we say that the LSR completed its restart.

The following procedures apply when an LSR is in the process of restarting.

6.1.1. Non-egress LSR

If the label carried in the Mapping message is not an Implicit NULL, the LSR searches its MPLS forwarding table for an entry with the outgoing label equal to the label carried in the message, and the next hop equal to one of the addresses (next hops) received in the Address message from the peer. If such an entry is found, the LSR no longer marks the entry as stale. In addition, if the entry is of type <incoming label, (outgoing label, next hop)> (rather than <FEC, (outgoing label, next hop)>), the LSR associates the incoming label from that entry with the FEC received in the Label Mapping message, and advertises (via LDP) <incoming label, FEC> to its neighbors. If the found entry has no incoming label, or if no entry is found, the LSR follows the normal LDP procedures. (Note that this paragraph describes the scenario where the restarting LSR is neither the egress, nor the penultimate hop that uses penultimate hop popping for a particular LSP. Note also that this paragraph covers the case where the restarting LSR is the ingress.)

If the label carried in the Mapping message is an Implicit NULL label, the LSR searches its MPLS forwarding table for an entry that indicates Label pop (means no outgoing label), and the next hop equal to one of the addresses (next hops) received in the Address message from the peer. If such an entry is found, the LSR no longer marks the entry as stale, the LSR associates the incoming label from that entry with the FEC received in the Label Mapping message from the neighbor, and advertises (via LDP) <incoming label, FEC> to its neighbors. If the found entry has no incoming label, or if no entry is found, the LSR follows the normal LDP procedures. (Note that this paragraph describes the scenario where the restarting LSR is a penultimate hop for a particular LSP, and this LSP uses penultimate hop popping.)

The description in the above paragraph assumes that the restarting LSR generates the same label for all the LSPs that terminate on the same LSR (different from the restarting LSR), and for which the restarting LSR is a penultimate hop. If this is not the case, and the

restarting LSR generates a unique label per each such LSP, then the LSR needs to preserve across the restart not just <incoming label, (outgoing label, next hop)> mapping, but also the FEC associated with this mapping. In such case the LSR would search its MPLS forwarding state for an entry that (a) indicates Label pop (means no outgoing label), (b) the next hop equal to one of the addresses (next hops) received in the Address message from the peer, and (c) has the same FEC as the one received in the Label Mapping message. If such an entry is found, the LSR no longer marks the entry as stale, the LSR associates the incoming label from that entry with the FEC received in the Label Mapping message from the neighbor, and advertises (via LDP) <incoming label, FEC> to its neighbors. If the found entry has no incoming label, or if no entry is found, the LSR follows the normal LDP procedures.

6.1.2. Egress LSR

If an LSR determines that it is an egress for a particular FEC, the LSR is configured to generate a non-NULL label for that FEC, and the LSR is configured to generate the same (non-NULL) label for all the FECs that share the same next hop and for which the LSR is an egress, the LSR searches its MPLS forwarding table for an entry that indicates Label pop (means no outgoing label), and the next hop equal to the next hop for that FEC. (Determining the next hop for the FEC depends on the type of the FEC. For example, when the FEC is an IP address prefix, the next hop for that FEC is determined from the IP forwarding table.) If such an entry is found, the LSR no longer marks this entry as stale, the LSR associates the incoming label from that entry with the FEC, and advertises (via LDP) <incoming label, FEC> to its neighbors. If the found entry has no incoming label, or if no entry is found, the LSR follows the normal LDP procedures.

If an LSR determines that it is an egress for a particular FEC, the LSR is configured to generate a non-NULL label for that FEC, and the LSR is configured to generate a unique label for each such FEC, then the LSR needs to preserve across the restart not just <incoming label, (outgoing label, next hop)> mapping, but also the FEC associated with this mapping. In such case the LSR would search its MPLS forwarding state for an entry that indicates Label pop (means no outgoing label), and the next hop equal to the next hop for that FEC associated with the entry (Determining the next hop for the FEC depends on the type of the FEC. For example, when the FEC is an IP address prefix, the next hop for that FEC is determined from the IP forwarding table.) If such an entry is found, the LSR no longer marks this entry as stale, the LSR associates the incoming label from that entry with the FEC, and advertises (via LDP) <incoming label, FEC> to its neighbors. If the found entry has no incoming label, or if no

entry is found, the LSR follows the normal LDP procedures.

If an LSR determines that it is an egress for a particular FEC, and the LSR is configured to generate a NULL (either Explicit or Implicit) label for that FEC, the LSR just advertises (via LDP) such label (together with the FEC) to its neighbors.

6.2. Alternative procedures for the restarting LSR

In this section we describe an alternative to the procedures described in Section 6.1.

The procedures described in this section assumes that the restarting LSR has (at least) as many unallocated as allocated labels. The latter form the MPLS forwarding state that the LSR managed to preserve across the restart.

After an LSR restarts its control plane, the LSR should check whether it was able to preserve its MPLS forwarding state from prior to the restart. If no, then the LSR must set the Recovery Time to 0 in the Graceful Restart TLV the LSR sends to its neighbors.

If the forwarding state has been preserved, then the LSR starts its internal timer, called MPLS Forwarding State Holding timer (the value of that timer should be configurable), and marks all the MPLS forwarding state entries as "stale". At the expiration of the timer, all the entries still marked as stale should be deleted. The value of the Recovery Time advertised in the Graceful Restart TLV should be set to the (current) value of the timer at the point when the Initialization message carrying the Graceful Restart TLV is sent.

We say that an LSR is in the process of restarting when the MPLS Forwarding State Holding timer is not expired. Once the timer expires, we say that the LSR completed its restart.

While an LSR is in the process of restarting, the LSR creates local label binding by following the normal LDP procedures.

Note that while an LSR is in the process of restarting, the LSR may have not one, but two local label bindings for a given FEC - one that was retained from prior to restart, and another that was created after the restart. Once the LSR completes its restart, the former will be deleted. Both of these bindings though would have the same outgoing label (and the same next hop).

6.3. Restart of LDP communication with a neighbor LSR

When an LSR detects that its LDP session with a neighbor went down, and the LSR knows that the neighbor is capable of preserving its MPLS forwarding state across the restart (as was indicated by the Graceful Restart TLV in the Initialization message received from the neighbor), the LSR should retain the label-FEC bindings received via that session (rather than discarding the bindings), but should mark them as "stale".

After detecting that the LDP session with the neighbor went down, the LSR should try to re-establish LDP communication with the neighbor.

The amount of time the LSR should keep its stale label-FEC bindings is set to the lesser of the Restart Time, as was advertised by the neighbor, and a local timer. After that, if the LSR still doesn't establish an LDP session with the neighbor, all stale bindings should be deleted. The local timer is started when the LSR detects that its LDP session with the neighbor went down. The value of the local timer should be configurable.

If the LSR re-establishes an LDP session with the neighbor within the lesser of the Restart Time and the local timer, and the LSR determines that the neighbor was not able to preserve its MPLS forwarding state, the LSR should immediately delete all the stale label-FEC bindings received from that neighbor. If the LSR determines that the neighbor was able to preserve its MPLS forwarding state (as was indicated by the non-zero Recovery Time advertised by the neighbor), the LSR should further keep the stale label-FEC bindings received from the neighbor for as long as the Recovery Time that the LSR advertises to the neighbor (after that, the bindings still marked as stale should be deleted). The Recovery Time that the LSR advertises to the neighbor should be greater than the Recovery Time the (restarting) neighbor advertised to the LSR.

The LSR should try to complete the exchange of its label mapping information with the neighbor within the Recovery Time, as specified in the Graceful Restart TLV received from the neighbor.

The LSR should handle the Label Mapping messages received from the neighbor by following the normal LDP procedures, except that (a) it should treat the stale entries in its Label Information Base (LIB), as if these entries have been received over the (newly established) session, (b) if the label-FEC binding carried in the message is the same as the one that is present in the LIB, but is marked as stale, the LIB entry should no longer be marked as stale, and (c) if for the FEC in the label-FEC binding carried in the message there is already a label-FEC binding in the LIB that is marked as stale, and the label

in the LIB binding is different from the label carried in the message, the LSR should just update the LIB entry with the new label.

An LSR, once it creates a <label, FEC> binding, should keep the value of the label in this binding for as long as the LSR has a route to the FEC in the binding. If the route to the FEC disappears, and then re-appears again later, then this may result in using a different label value, as when the route re-appears, the LSR would create a new <label, FEC> binding. To minimize the potential mis-routing caused by the label change, when creating a new <label, FEC> binding the LSR should pick up the least recently used label. Once an LSR releases a label, the LSR should not re-use this label for advertising a <label, FEC> binding to a neighbor that supports graceful restart for at least the sum of Restart Time plus Recovery Time, as advertised by the neighbor to the LSR.

7. Security Consideration

This document does not introduce new security issues. The security considerations pertaining to the original LDP protocol remain relevant.

8. Intellectual Property Considerations

Juniper Networks, Inc. is seeking patent protection on some or all of the technology described in this Internet-Draft. If technology in this document is adopted as a standard, Juniper Networks agrees to license, on reasonable and non-discriminatory terms, any patent rights it obtains covering such technology to the extent necessary to comply with the standard.

Redback Networks, Inc. is seeking patent protection on some of the technology described in this Internet-Draft. If technology in this document is adopted as a standard, Redback Networks agrees to license, on reasonable and non-discriminatory terms, any patent rights it obtains covering such technology to the extent necessary to comply with the standard.

9. Acknowledgments

We would like to thank Chaitanya Kodeboyina, Nischal Sheth, and Enke Chen for their contributions to this document.

10. References

[LDP] "Label Distribution Protocol", RFC3036

[OSPF-RESTART] "Hitless OSPF Restart", draft-ietf-ospf-hitless-restart-01.txt

[ISIS-RESTART] "Restart signaling for ISIS", draft-shand-isis-restart-00.txt

[BGP-RESTART] "Graceful Restart Mechanism for BGP", draft-ietf-idr-restart-00.txt

11. Author Information

Manoj Leelanivas
Juniper Networks
1194 N.Mathilda Ave
Sunnyvale, CA 94089
e-mail: manoj@juniper.net

Yakov Rekhter
Juniper Networks
1194 N.Mathilda Ave
Sunnyvale, CA 94089
e-mail: yakov@juniper.net

Rahul Aggarwal
Redback Networks
350 Holger Way
San Jose, CA 95134
e-mail: rahul@redback.com