LSR WG
Internet-Draft
Intended status: Standards Track
Expires: February 20, 2020

Shaofu. Peng
Zheng. Zhang
ZTE Corporation

August 19, 2019

IGP Flooding Optimization Methods
draft-peng-lsr-igp-flooding-opt-methods-00

Abstract

This document mainly describe a method to optimize IGP flooding by visited record, the visited record information could be encapsulated in outer carring header, or as a part of IGP PDU.

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 https://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 February 20, 2020.

Table of Contents

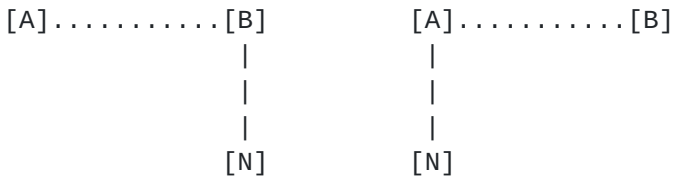
1. Introduction 2
2. Solutions begin First Established Phase 3
2.1. BIER based IGP flooding 4
2.1.1. Overview 4
2.1.2. BIER Encapsulation Extensions 4
2.1.3. IGP Capability Extensions 5
2.1.4. Operations 5
2.1.4.1. Local Generated Link State Data 5
2.1.4.2. Remote Generated Link State Data 5
2.1.4.3. Not Directly Connected Neighbors in Tier-based Networks 6
2.1.4.4. Error Correction 7
2.1.5. Other considerations 7
2.1.6. Examples 7
2.1.6.1. A Sparse Network Example 8
2.1.6.2. A Tier-based Densy Network Example 9
2.1.6.3. A Fullmesh Densy Network Example 10
2.2. IGP Extensions to Record Visited Nodes 11
3. Solutions after First Established Phase 11
4. Security Considerations 11
5. IANA Considerations 11
6. Normative References 11
Authors' Addresses 12

1. Introduction

IGP flooding issue of densy networks such as spine-leaf, Clos, or Fat Tree topology has get creased attentions and solution seeking. Conventional IS-IS, OSPFv2 and OSPFv3 all perform redundantly flooding information throughout the dense topology, leading to overloaded control plane inputs and thereby creating operational issues.

[I-D.ietf-lsr-dynamic-flooding] has ananylized the issues and described a common solution to build a sparse FT (Flooding Topology) dedicated to link state packet flooding. However it is a bit complex to cover all sceneries to compute an optimal FT to reduce the redundancy flooding, sometimes it need a rollback to traditional flooding rules to guarantee function correct and have to abandon performance. Implementors have to consider too many type of events that maybe affect the FT based flooding behavior with special careful detail treatment per specific event. For example, in some cases both a new FT and an old FT need work together, in some cases a temporary flooding on non-FT link is needed.

The following figure 1 simply illustrate a possilbe timing sequence example according to FT solution. Although we believe it can be easily addressed, it just indicates the inherent complexity of this solution that must be given adequate care.



(a) FT on node A (b) FT on node B

Fig.1 FT inconsistency on multiple nodes

Suppose at some time node A computed the FT as Fig.1(a), node B computed the FT as Fig.1(b), this inconsistency would be eliminated at last, but just at this time, a link state data need be flooded along FT, so node A thought node B would propagate data to N, but node B also thought node A would propagate data to N, the result is that nobody propagated data to N.

Note the FT itself need to be computed frequently triggered by any topology events, especially during the first established phase of network, where the ultimal optimal FT can be computed just based on the full stable topology database that maybe hard to get from the fully redundant flooding. The computation overhead maybe offset its benefits.

This document try to discuss some other possible methods to optimize IGP flooding with little cost, simple logic, and implementation friendly.

2. Solutions begin First Established Phase

Netwrok administrator expect to solve the redundant flooding problem from the beginning of a densy network power-on, to quickly deploy service, it can't tolerate a long time to get a stable network.

A solution maybe possible to record the potential visited node of the link state data packet, to filter nodes that have already been visited. We will discuss two methods to record the visited nodes as following.

2.1. BIER based IGP flooding

2.1.1. Overview

Bit Index Explicit Replication (BIER) [RFC8279] is an architecture that provides optimal multicast forwarding without requiring intermediate routers to maintain any per-flow state by using a multicast-specific BIER header. [RFC8296] defines two types of BIER encapsulation formats: one is MPLS encapsulation, the other is non-MPLS encapsulation. It is convenient to use BIER to record visited nodes. To fulfill IGP flooding optimization, some extensions need be applied to BIER encapsulation.

For an IGP area/level, a BIER sub-domain is used to construct the IGP tolology. Supposed that each node in the IGP area/level is BIER-enabled, they belong to the same BIER sub-domain. Each node is provisioned with a "BFR-id" that is unique within the sub-domain. Now a "BIER Record" function is introduced to BIER forwarding mechanism defined in [RFC8279] and [RFC8296]. "BIER Record" function will record the BIER packet, which contains the IGP link state data such as ISIS LSP(Link State PDU) or OSPF LSU(Link State Update), has visited how many nodes, i.e, the bit-string included in the BIER header of a "BIER Record" packet will contain the related BP(bit position) of all visited nodes' BFR-id. Once a node received a link state data contained in "BIER Record" packet, it never continues to flood the data toward to the neighbors that have already existed in the received bit-string.

2.1.2. BIER Encapsulation Extensions

[RFC8296] defines the BIER encapsulation format, the "Rsv" field is currently unused, a new bit (the rightmost bit) of the "Rsv" field can be used for flag-R (Record), if set to 1 indicate the BIER packet is a "BIER Record" packet, otherwise is a traditional BIER packet. "BIER Record" packet received on a node can never be forwarded again, the TTL field in the received "BIER Record" packet MUST be always set to 1.

The "Proto" field is currently not provided to encapsulate IGP payload. IANA has assigned value 1~6 for "Proto" field, a new value (suggested 7) is to indicate the encapsulated payload is ISIS LSP(Link State PDU), a new value (suggested 8) is to indicate the encapsulated payload is OSPF LSU(Link State Update).

2.1.3. IGP Capability Extensions

Each node inside the IGP area/level can be provisioned whether or not support BIER based IGP flooding capability and advertised this router capability to other nodes.

A new flag (flag-B) is introduced for Flags field of IS-IS Router Capability TLV-242 [RFC7981] as well as Informational Capabilities of OSPF Router Informational Capabilities TLV [RFC7770], if set to 1 indicate the advertised node has BIER based IGP flooding capability, otherwise has not.

2.1.4. Operations

2.1.4.1. Local Generated Link State Data

Suppose that a node A generates a link state data, e.g, because of a new link inserting, it will flood the data (ISIS LSP or OSPF LSU) to neighbor N. If both A and N support BIER based IGP flooding capability, node A can send the data contained in the "BIER Record" packet to node N, the send-bitstring, i.e, the bit-string of BIER header of the sending "BIER Record" packet, will include BP of A and all its neighbors (including N). Note that if there are multiple links between A and N, only one link is choosen to send packet.

If any of node A and N can not support BIER based IGP flooding capability, node A will take the traditional flooding mechanism to flood data to N, i.e, the link state data is not encapsulated in BIER header but in traditional L2 header (for ISIS) or IP header (for OSPF).

Network administrator can config local policy on all nodes in the network to force to send link state data by "BIER Record" packet if he ensure that all nodes are really capable of BIER based IGP flooding. This policy is useful to speed up the convergence during the early phase of network power on.

2.1.4.2. Remote Generated Link State Data

Node A can also receive a remote link state data from neighbor N, the data maybe originated from N itself or a third node. The data could be received by traditional IGP flooding mechanism or "BIER Record" packet (we term the bit-string of BIER header of the received "BIER Record" packet as recv-bitstring).

In former case, node A will check if there are already an item with the same KEY existed in the local LSDB and compare who is new and who is old. If no local item or local item is old, node A need add or

update the data to local LSDB, and continue to flood it towards other neighbors except N. If local item is new, node A just flood the local item to neighbor N. If local item is totally same as received data, no processing.

In later case, node A MUST drop the received data if it has not BIER based IGP flooding capability, otherwise it will also check if there are already an item with the same KEY existed in the local LSDB and compare who is new and who is old. If no local item or local item is old, node A need add or update the data to local LSDB, and continue to flood it towards other neighbors except N and neighbors contained in recv-bitstring. If local item is new, node A just flood the local item to neighbor N. If local item is totally same as received data, no processing.

2.1.4.2.1. Continuous Flooding Procedure

For the above two cases, if node A need continue to flood the remote link state data to any neighbors, it need check If both itself and the neighbor support BIER based IGP flooding capability, if yes node A can send the data contained in the "BIER Record" packet to the neighbor, the send-bitstring will include BP of A, and all neighbors of A, and all nodes already contained in recv-bitstring especially for the above later case.

If any of node A and the neighbor can not support BIER based IGP flooding capability, node A will take the traditional flooding mechanism to flood data to the neighbor, i.e, the link state data is not encapsulated in BIER header but in traditional L2 header (for ISIS) or IP header (for OSPF).

2.1.4.3. Not Directly Connected Neighbors in Tier-based Networks

Data centers often deployed a spine-leaf, Clos, or Fat Tree topology, the key feature is that this class of topology is constructed with serveral tiers, nodes in the same tier have connections rarely, but each node have full connections to all nodes in the neighbor tier.

Although a node A within tier-x has not any connections with other nodes in the same tier-x, it can configure local policy to preserve these not-directly connected (NDC) neighbors. These NDC neighbors within same tier can be explicitly inserted to the send-bitstring of "BIER Record" packets towards most of real neighbor nodes in the neighbor tier, but a very few neighbor (extremely a single neighbor) in the neighbor tier received the "BIER Record" packe without NDC neighbors inserting. This policy can significantly reduce the redundant flooding.

2.1.4.4. Error Correction

As a node decides whether or not to flood remote link state data to a neighbor according to the recv-bitstring, some neighbors that not be included in recv-bitstring will receive the data, some other neighbors that be included in recv-bitstring will be filtered and can not receive the data. In extremity, the filtered neighbors maybe exactly not receive the data before, due to link interrupt.

The same issue can be occurred for local link state data, the data will send to all neighbors with send-string including all neighbors, but one neighbor maybe exactly not receive the data due to link interrupt right now.

To recovery the lost data toward a neighbor, normal database synchronization mechanisms (i.e., OSPF DDP, IS-IS CSNP) would apply between local and remote node. Traditionally, database synchronization packet is periodicaly sent on broadcast link to confirm that all nodes connected to the LAN has the same LSDB. This document adjust it to any type of link. As long as the node enable the capability of BIER based IGP flooding, it will apply the database synchronization mechanism with neighbor in despite of the type of link between them. For broadcast link, a DR or DIS is elected to send the database synchronization packet periodicaly. For P2P link, similar election method could be used to let one side with high priority to send the database synchronization packet. The period is recommended long.

Due to database synchronization mechanism, if any link state data need to be flooded from one side to another, the operations are totally similar with section "2.1.4.2.1 Continuous Flooding Procedure".

2.1.5. Other considerations

As defined in [RFC8401] and [RFC8444], BFR-id is advertised within prefix reachability, that would be too late for a node to get the BFR-id information of all neighbors when a link state data is launched. A new advertisement method maybe that each node carry local BFR-id in IGP hello packets, if this is true non-MPLS BIER encapsulation is suitable.

2.1.6. Examples

2.1.6.1. A Sparse Network Example

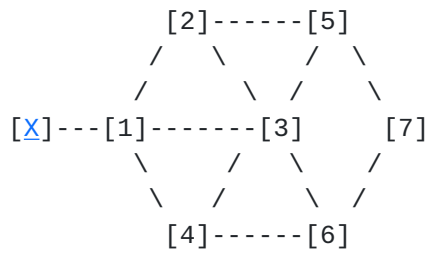


Fig.2 A Sparse Network Example

Fig.2 shows a sparse network, which is constructed by node 1~7 and the corresponding links originally. Now a new node X with its link is added to the network. Suppose that all nodes have BIER based IGP flooding capability.

From the perspective of node 1, it will create session with node X, and an local link state data for unidirectional link(1->X) is generated, node 1 will send it by "BIER Record" packet with send-bitstring (X, 1, 2, 3, 4) toward each neighbor, i.e, X, 2, 3, 4.

Node 2 receives the "BIER Record" packet from node 1, extracte the link state data from the packet and store it in local LSDB. Because the recv-bitstring has already contained neighbor 3, node 2 just continues to flood the data to neighbor 5, i.e, a new "BIER Record" packet is produced with send-bitstring (X, 1, 2, 3, 4, 5) which is combined with node 2 itself, plus all neighbors of node 2, plus all nodes already in recv-bitstring.

Similarly, Node 5 receives the data from node 2, store it in local LSDB and only continues to flood it to node 7. Node 5 will receive the data from node 3 repeatedly, because node 3 will also receive data from node 1 with recv-bitstring (X, 1, 2, 3, 4) without 5.

Node 6 is mostly like node 5.

Note that node X will also generate a local link state data for unidirectional link(X->1), no further elaboration.

Also note that the new node X will receive stock link state data from node 1 according to normal database synchronization immediately caused by link UP.

In this example, we can see that the redundant flooding behavior is suppressed within limits, as redundant flooding behavior in sparse network is not serious at all.

2.1.6.2. A Tier-based Densy Network Example

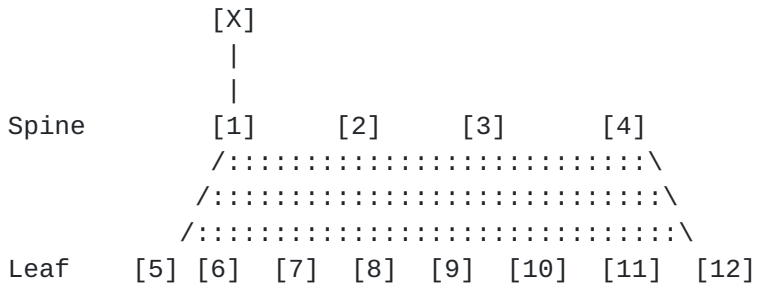


Fig.3 A Tier-based Densy Network

Fig.3 shows a Spine-leaf densy network, which is constructed by node 1~12 and the corresponding links originally. Node 1~4 is within spine tier, node 5~12 is within leaf tier. Each spine node connects all leaf nodes, and vice versa. Now a new node X with its link is added to the network. Suppose that all nodes have BIER based IGP flooding capability.

From the perspective of node 1, it will create session with node X, and an local link state data for unidirectional link(1->X) is generated. As above mentioned, although node 1 within tier-spine has not any connections with other nodes (2, 3, 4) in the same tier-spine, it can configure local policy to preserve these not-directly connected (NDC) neighbors. So node 1 will send the link state data by "BIER Record" packet with send-bitstring (X, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12) including NDC neighbors toward most of neighbors in tier-leaf, i.e, 6~12, but send "BIER Record" packet with send-bitstring (X, 1, 5, 6, 7, 8, 9, 10, 11, 12) toward a single neighbor in tier-leaf, i.e, 5. Note that node 5 must be an active node, if not a new single neighbor in tier-leaf must be selected to receive data without NDC neighbors inserting.

Node 5 receives the "BIER Record" packet from node 1, extracte the link state data from the packet and store it in local LSDB. Node 5 continues to flood the data to neighbor 2, 3, 4, i.e, a new "BIER Record" packet is produced with send-bitstring (X, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12) which is combined with node 5 itself, plus all neighbors of node 5, plus all nodes already in recv-bitstring.

Node 2 receives the "BIER Record" packet from node 5, extracte the link state data from the packet and store it in local LSDB. Because the recv-bitstring has already contained all neighbors, node 2 no longer continues to flood the data.

Node 3, 4 is similar to node 2.

Node 6 receives the "BIER Record" packet from node 1, extracte the link state data from the packet and store it in local LSDB. Because the recv-bitstring has also contained all neighbors (due to NDC neighbors inserting), node 6 no longer continues to flood the data.

Node 7~12 is similar to node 6.

In this example, we can see that the redundant flooding behavior is suppressed with a definite improvement, other densy tier-based networks have the same optimizing effect.

2.1.6.3. A Fullmesh Densy Network Example

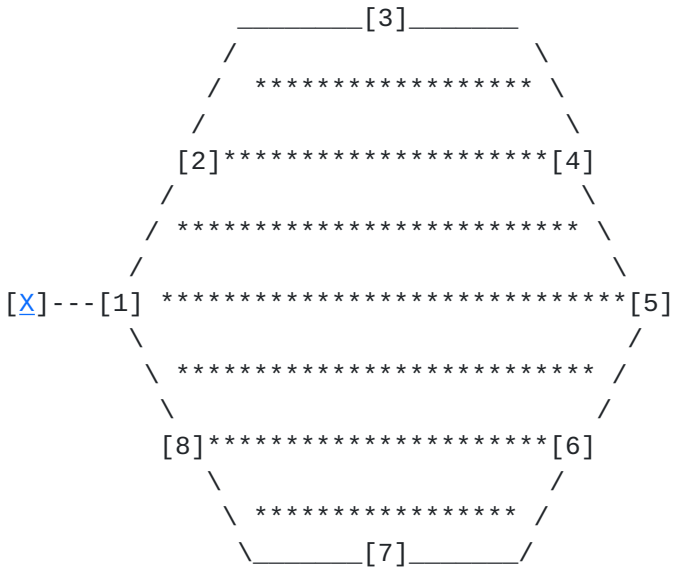


Fig.4 A Fullmesh Densy Network

Fig.4 shows a fullmesh densy network, which is constructed by node 1~8 and the corresponding links originally. Each node directly connects all other nodes. Now a new node X with its link is added to the network. Suppose that all nodes have BIER based IGP flooding capability.

How the optimization is reached is just like example 1, but in this example we will see the local link state data that generated on node 1 will never continue to be flooded again by any other receiving nodes, the redundant flooding behavior is suppressed completely.

2.2. IGP Extensions to Record Visited Nodes

Although BIER is convenient to carry potential visited nodes information of link state data, some network may not delpoy BIER. Alternate method is to directly extend ISIS or OSPF protocol to carry visited nodes information that is advertised with ISIS LSP or OSPF LSU.

The troublesome problem is that according to traditonal ISIS LSP or OSPF LSU packet processing rules, the content of these type of packets can not be changed by transient nodes otherwise multiple copy with the same KEY but different content (e.g, different visited nodes information) received on a node will cause a checksum error. So the visited nodes information MUST not be included for checksum computation and MUST not be stored in LSDB for path computation, it is only used for flooding control.

The detailed extensions for ISIS and OSPF will be discussed in the next version of this document.

3. Solutions after First Established Phase

Netwrok administrator maybe let go of the redundant flooding behavior during first established phase of network power-on, but seek solutions to supress the subsequent redundant flooding after the network is stable.

Each node could have a waiting period to act as traditional flooding behavior, when the waiting timer expired it will act as enhanced flooding behavior.

The possible methods will be discussed in the next version of this document.

4. Security Considerations

TBD

5. IANA Considerations

TBD

6. Normative References

[I-D.ietf-lsr-dynamic-flooding]
Li, T., Psenak, P., Ginsberg, L., Chen, H., Przygienda, T., Cooper, D., Jalil, L., and S. Dontula, "Dynamic Flooding on Dense Graphs", draft-ietf-lsr-dynamic-flooding-03 (work in progress), June 2019.

[RFC7770] Lindem, A., Ed., Shen, N., Vasseur, JP., Aggarwal, R., and S. Shaffer, "Extensions to OSPF for Advertising Optional Router Capabilities", RFC 7770, DOI 10.17487/RFC7770, February 2016, <https://www.rfc-editor.org/info/rfc7770>.

[RFC7981] Ginsberg, L., Previdi, S., and M. Chen, "IS-IS Extensions for Advertising Router Information", RFC 7981, DOI 10.17487/RFC7981, October 2016, <https://www.rfc-editor.org/info/rfc7981>.

[RFC8279] Wijnands, IJ., Ed., Rosen, E., Ed., Dolganow, A., Przygienda, T., and S. Aldrin, "Multicast Using Bit Index Explicit Replication (BIER)", RFC 8279, DOI 10.17487/RFC8279, November 2017, <https://www.rfc-editor.org/info/rfc8279>.

[RFC8296] Wijnands, IJ., Ed., Rosen, E., Ed., Dolganow, A., Tantsura, J., Aldrin, S., and I. Meilik, "Encapsulation for Bit Index Explicit Replication (BIER) in MPLS and Non-MPLS Networks", RFC 8296, DOI 10.17487/RFC8296, January 2018, <https://www.rfc-editor.org/info/rfc8296>.

[RFC8401] Ginsberg, L., Ed., Przygienda, T., Aldrin, S., and Z. Zhang, "Bit Index Explicit Replication (BIER) Support via IS-IS", RFC 8401, DOI 10.17487/RFC8401, June 2018, <https://www.rfc-editor.org/info/rfc8401>.

[RFC8444] Psenak, P., Ed., Kumar, N., Wijnands, IJ., Dolganow, A., Przygienda, T., Zhang, J., and S. Aldrin, "OSPFv2 Extensions for Bit Index Explicit Replication (BIER)", RFC 8444, DOI 10.17487/RFC8444, November 2018, <https://www.rfc-editor.org/info/rfc8444>.

Authors' Addresses

Shaofu Peng
ZTE Corporation
No.68 Zijinghua Road,Yuhuatai District
Nanjing 210012
China

Email: peng.shaofu@zte.com.cn

Zheng(Sandy) Zhang
ZTE Corporation
No.50 Software Avenue,Yuhuatai District
Nanjing 210012
China

Email: zzhang_ietf@hotmail.com