Workgroup: Internet Engineering Task Force
Internet-Draft: draft-ietf-lsr-isis-ttz-09
Published: 28 March 2024
Intended Status: Experimental
Expires: 29 September 2024

Authors: H. Chen (Futurewei), R. Li (Futurewei), Y. Yang (IBM), A. Kumar S N (RtBrick), Y. Fan (Casa Systems), N. So, V. Liu, M. Toy (Verizon), L. Liu (Fujitsu), K. Makhijani (Futurewei)

IS-IS Topology-Transparent Zone

Abstract

This document specifies a topology-transparent zone in an IS-IS area. A zone is a subset (block/piece) of an area, which comprises a group of routers and a number of circuits connecting them. It is abstracted as a virtual entity such as a single virtual node or zone edges mesh. Any router outside of the zone is not aware of the zone. The information about the circuits and routers inside the zone is not distributed to any router outside of the zone.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

[ISO10589] and [RFC1195] describe two levels of areas in IS-IS, level 1 and level 2 areas. There are scalability issues in using areas as the number of routers in a network becomes larger and larger. When an IS-IS area becomes larger, its convergence on a network event such as a link down will take a longer time. During the period of network converging, more traffic that is transported through the network area will get lost.

Through splitting the network into multiple level 1 areas connected by level 2, we may extend the network further. However, dividing a network from one area into multiple areas or from a number of existing areas to even more areas can be a challenging and time consuming task since it involves significant network architecture changes. It needs a careful planning and many configurations on the network.

These issues can be resolved by using topology-transparent zone (TTZ), which abstracts a zone (i.e., a subset of an area) as a single virtual node or zone edges' mesh with minimum efforts and minimum service interruption. Note that a zone can be an entire area.

This document presents topology-transparent zone and specifies extensions to IS-IS that support topology-transparent zone.

1.1. 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] [RFC8174] when, and only when, they appear in all capitals, as shown here.

1.2. Terminology

Zone: A subset (block or piece) of an area. In a special case, a zone is an entire area.

TTZ: Topology-Transparent Zone (TTZ) is a mechanism that abstracts a zone as a single virtual node or zone edges' full mesh. The virtual node appears connected to all the zone neighbors.

TTZ Virtual Entity: A single virtual node or zone edges' full mesh to which a zone is transformed using TTZ.

A TTZ: A zone that is (to be) abstracted using TTZ.

Zone External Node: A node outside of a zone.

Zone Internal Node: A node within a zone without any connection to a node outside of the zone.

Zone Edge/Border Node: A node that is part of a zone connecting to a node outside of the zone.

Zone Node: A zone internal node or a zone edge/border node (i.e., a node that is part of a zone).

Zone Link: A link connecting zone nodes (i.e., a link that is part of a zone).

Zone Neighbor Node: A node outside of a zone that is a neighbor of a zone edge/border node.

Zone Neighbor: A Zone Neighbor Node.

CLI: Command Line Interface.

LSP: A Link State Protocol Data Unit (PDU) in IS-IS. An LSP contains link state information. In general, a router/node originates multiple LSPs, distinguished by LSP fragment number, to carry the link state information about it and the links attached to it.

LS: Link State. In general, the LS for a node is all the LSPs that the node originates. The LS for a zone is the set of LSPs that all the nodes in the zone originate to carry the information about them and the links attached to them inside the zone.

2. Requirements

Topology-Transparent Zone (TTZ) may be deployed to resolve some critical issue of scalability in existing networks and future networks. The requirements for TTZ are as follows:

* TTZ MUST be backward compatible. When a TTZ is deployed on a set of routers in a network, the routers outside of the TTZ in the network do not need to know or support TTZ.

* TTZ MUST support at least one more levels of network hierarchy, in addition to the hierarchies supported by existing IS-IS.

* Transforming a zone (i.e., a block of network area) to a TTZ virtual entity SHOULD be smooth with minimum service interruption. A TTZ virtual entity is either a single virtual node or zone edges' full mesh.

* Transforming (or say rolling back) a TTZ virtual entity back to its zone (i.e., its original block of network area not using TTZ) (refer to Section 5.2) SHOULD be smooth with minimum service interruption.

* The configuration for a TTZ in a network SHOULD be minimal.

* The changes on the existing protocols to support TTZ SHOULD be minimal.

3. Zone Abstraction

When abstracting a zone, a user may select one of two models: node abstraction model and mesh abstraction model.

3.1. Node Abstraction Model

In node abstraction model (or node model for short), a zone is abstracted as a single virtual node. The virtual node represents the entire zone. It appears connected to all the zone neighbors and is in the same area as those neighbors.

Deploying node model may cause changes on some routes since the block of an area (zone) becomes a single virtual node. Some of the routes that are optimal before the abstraction may be changed to be suboptimal after the abstraction (refer to Section 4.1.5). This may attract traffic to the TTZ and change the balance of traffic in the network.

The advantage of node model is that it provides a higher degree of abstraction rate than the mesh model. It is more scalable.

3.2. Mesh Abstraction Model

In mesh abstraction model (or mesh model for short), a zone is abstracted as its edges' full mesh [RFC8099], there is a full mesh of connections among the edges and each edge is also connected to its neighbors outside of the zone.

The advantage of mesh model is that it keeps the routes unchanged. After a zone is abstracted as the full mesh of the edges of the zone, every route is still optimal (refer to Section 4.3.3).

The disadvantage of mesh model is that it does not scale when the number of edge nodes of a zone is large.

4. Topology-Transparent Zone

A Topology-Transparent Zone (TTZ) comprises an Identifier (ID) and a subset (piece/block) of an area such as a Level 2 area in IS-IS. It is abstracted as a single virtual node or its edges' full mesh. TTZ and zone as well as node and router will be used interchangeably below.

A zone MUST be within a single area. In addition, all the nodes in a zone MUST reside within a common level. There are three cases. All the nodes in a zone are L1 nodes except for some of edge nodes of the zone may be L1/L2 nodes. All the nodes in a zone are L2 nodes except for some of edge nodes of the zone may be L1/L2 nodes. All the nodes in a zone are L1/L2 nodes.

4.1. Zone as a Single Node

After a zone is abstracted as a single virtual node having a virtual node ID, every node outside of the zone sees a number of links connected to this single node. Each of these links connects to a zone neighbor. The link states inside the zone are not advertised to any node outside of the zone. The virtual node ID may be derived from the zone ID. The value of the zone ID is transferred to four bytes of an IPv4 address, and then to 12 digitals of the IPv4 address in dotted form. The node ID of 6 bytes is from these 12 digitals, 2 digitals for 1 byte.

The sections below describe the behaviors of zone nodes when/after a zone is abstracted to a single virtual node. They are summarized as follows.

* Zone leader originates the LS (i.e., a set of LSPs) for the virtual node (refer to Section 4.1.3).

* Zone nodes re-advertise the LS originated by the zone leader (refer to Section 4.1.3 and Section 4.1.4).

* Zone edge/border node forms adjacencies with zone neighbor nodes using the identity of the virtual node not its own identity (refer to Section 4.1.4).

* Zone edge/border node re-advertises the LS for the virtual node as it originates the LS (refer to Section 4.1.4).

* Zone edge/border node purges its existing LSP and originates a new LSP containing its zone links after receiving the LS for the virtual node (refer to Section 4.1.4).

* Zone edge/border nodes do not advertise the LSPs originated by zone nodes to its zone neighbors (refer to Section 4.1.4 and Section 4.4.1).

* Zone edge/border nodes continue to operate IS-IS as normal to advertise the LSPs received from its zone neighbors (refer to Section 4.1.4 and Section 4.4.2).

* Zone internal nodes continue to operate IS-IS as normal to advertise the LSPs received from its neighbors (refer to Section 4.4.1).

* Zone nodes compute routes from the topology without the virtual node (refer to Section 4.1.5).

4.1.1. An Example of Zone as a Single Node

The figure below shows an example of an area containing a TTZ: TTZ 600.

```
             TTZ 600
               \
                \ ^~^~^~^~^~^~^~^~^~^~^~^~
                 (                        )
===[R15]========(==[R61]------------[R63]==)======[R29]===
    ||\        (   |    \          /    |   )       ||
    || \       (   |     \        /     |   )       ||
    ||  \___   (   |      \      /      |   )       ||
    ||      \  (   |    ___\    /       |   )       ||
    ||       \ (   |   /   [R71]        |   )       ||
    ||        \(   | [R73] /    \       |   )       ||
    ||         (\  |      /      \      |   )       ||
    ||         ( \ |     /        \     |   )       ||
    ||         (  \|    /          \    |   )       ||
===[R17]========(==[R65]------------[R67]==)======[R31]===
     \\          (//                     \\)      //
      ||         //v~v~v~v~v~v~v~v~v~v~v~\\       ||
      ||        //                        \\      ||
      ||       //                          \\     ||
       \\     //                            \\  //
   ======[R23]==============================[R25]=====
         //                                     \\
        //                                       \\
```

Figure 1: An Example of TTZ 600

The area comprises routers R15, R17, R23, R25, R29 and R31. It also contains TTZ 600, which comprises routers R61, R63, R65, R67, R71 and R73, and the circuits connecting them.

There are two types of routers in a TTZ: TTZ internal routers and TTZ edge/border routers. A TTZ internal router is a router inside the TTZ and its adjacent routers are inside the TTZ. A TTZ edge/ border router is a router inside the TTZ and has at least one adjacent router that is outside of the TTZ.

The TTZ in the figure above comprises four TTZ edge/border routers R61, R63, R65 and R67. Each TTZ edge/border router is connected to at least one router outside of the TTZ. For instance, router R61 is a TTZ edge/border router since it is connected to router R15, which is outside of the TTZ.

In addition, the TTZ comprises two TTZ internal routers R71 and R73. A TTZ internal router is not connected to any router outside of the TTZ. For instance, router R71 is a TTZ internal router since it is not connected to any router outside of the TTZ. It is just connected to routers R61, R63, R65, R67 and R73 inside the TTZ.

A TTZ MUST hide the information inside the TTZ from the outside. It MUST NOT directly distribute any internal information about the TTZ to a router outside of the TTZ.

From a router outside of the TTZ, a TTZ is seen as a single node (refer to the Figure below). For instance, router R15, which is outside of TTZ 600, sees TTZ 600 as a single node Rz, which has normal connections to R15, R29, R17 and R23, R25 and R31.

```
              TTZ 600
                \
                 \ ^~^~^~^~^~^~^~^~^~^~^~^~
                  (                        )
 ===[R15]========(                          )======[R29]===
     ||\        (                            )       ||
     || \       (                            )       ||
     ||  \      (                            )       ||
     ||   \     (                            )       ||
     ||    \    (             Rz             )       ||
     ||     \   (                            )       ||
     ||      \  (                            )       ||
     ||       \ (                            )       ||
     ||        \(                            )       ||
 ===[R17]========(                          )======[R31]===
      \\          (                        )       //
       ||         //v~v~v~v~v~v~v~v~v~v~v~\\      ||
       ||        //                        \\     ||
       ||       //                          \\    ||
        \\     //                            \\  //
    ======[R23]==============================[R25]=====
          //                                     \\
         //                                       \\
```

Figure 2: TTZ 600 as Single Node Rz

4.1.2. Zone Leader Election

A node in a zone is elected as a leader for the zone, which is the node with the highest priority (and the highest node ID when there are more than one nodes having the same highest priority) in the zone. The leader election mechanism described in [I-D.ietf-lsr-dynamic-flooding] is used to elect the leader for the zone.

4.1.3. LS Generation for Zone as a Single Node

The leader for the zone originates the LS (i.e., set of LSPs) for the zone as a single virtual node and sends it to its neighbors. Each of the nodes in the zone re-advertises the LS to all its neighbors except for the one from which the LS is received.

This LS comprises all the adjacencies between the virtual node and the zone neighbors. The System ID of each LSP ID is the ID of the virtual node for the zone. The Source ID or Advertising Node/Router ID is the ID of the virtual node.

In addition, this LS may contain the IP prefixes such as the loopback IP addresses inside the zone to be accessed by zone external nodes (i.e., nodes outside of the zone). These IP prefixes are included in the IP internal reachability TLV.

When the existing zone leader fails, a new zone leader is elected. The new leader originates the LSPs for the virtual node based on the LSPs received from the failed leader. It retains the System ID of each LSP ID and the live adjacencies between the virtual node and the zone neighbors.

4.1.4. Adjacency Establishment

A zone edge node X, acting as a proxy for the single virtual node for the zone, forms a new adjacency between the virtual node and a node Y that is outside of the zone and in node X's area. There are two cases. One case is that there is an existing adjacency between X and Y; the other is that no adjacency exists between X and Y.

4.1.4.1. New Adjacency with Existing One

At first, zone edge node X acting as a proxy for the virtual node creates a new adjacency between the virtual node for the zone and node Y in a normal way. It sends Hellos and other packets containing the virtual node ID as Source ID to node Y. Node Y establishes an adjacency with the virtual node in the normal way.

Then, after receiving the LS for the virtual node originated by the zone leader, node X does a number of things as follows.

It terminates the existing adjacency between node X and node Y. It stops sending Hellos for the adjacency to node Y. Without receiving Hellos from node X for a given time such as hold-timer interval, node Y removes the adjacency to node X. Even though this adjacency terminates, node X keeps the link to node Y in its LSP.

It stops advertising or readvertising the LSPs that are originated by the zone nodes to node Y (also refer to Section 4.4.1).

It purges its current LSP and originates a new LSP containing its zone links. The new LSP does not contain the information about the adjacencies to the zone neighbors. It advertises the new LSP to its neighbors in the zone (also refer to Section 4.4.1). It does not advertise the new LSP to its zone neighbors.

It re-advertises the LS to all its neighbors except for the one from which the LS is received. It re-advertises the LS to node Y as it originates the LS.

It re-advertises the LSP received from zone neighbor Y to its other neighbors, including the nodes in the zone, which re-advertise the

LSP (received from Y outside of the zone) as normal IS-IS protocol operations (also refer to Section 4.4.2).

In the case where node Y is not in node X's area, is in the backbone and connected to node X, node X, acting as a proxy for the virtual node, creates a new adjacency between the virtual node and node Y in a normal way and sends the LS for the virtual node to node Y if the zone includes all the nodes in its area.

4.1.4.2. New Adjacency without Existing One

Every IS-IS protocol packet, such as Hello, that zone edge node X originates and sends node Y, uses the virtual node ID as Source ID.

When node X synchronizes its link state database (LSDB) with node Y, it sends Y all the link state information except for the link state belonging to the zone that is hidden from the nodes outside of the zone.

At the end of the LSDB synchronization, the LS for the zone as a single virtual node is originated by the zone leader and distributed to node Y. This LS contains the adjacencies between the virtual node and all the zone neighbors, including this newly formed zone neighbor Y.

Then node X has the same behaviors as those described above except for terminating the existing adjacency and purging its existing LSP.

4.1.5. Computation of Routes

After a zone is transferred/migrated to a single virtual node, every zone node computes the routes (i.e., shortest paths to the destinations) using the graph consisting of the zone topology, the connections between each zone edge and its zone neighbor, and the topology outside of the zone without the virtual node. The metric of a link outside of the zone is one order of magnitude larger than the metric of a link inside the zone.

Every node outside the zone computes the routes using the topology outside of the zone with the virtual node. The node does not have the topology inside the zone. The metric of every link outside of the zone is not changed.

4.2. Extensions to Protocols

This document defines a new TLV for use in IS-IS as follows.

* Zone ID TLV: containing a zone ID, a flags field and optional sub-TLVs.

4.2.1. Zone ID TLV

The format of IS-IS Zone ID TLV is illustrated below. It MUST be added into an LSP for a zone node.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Type (TBD1)  |     Length    |            Zone ID            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Zone ID (Continue)                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  RESV                |E|  OP |            Sub TLVs            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               +
~                                                               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: IS-IS Zone ID TLV

Type (1 byte): TBD1.

Length (1 byte): Its value is variable with a minimum of 8. A value larger than 8 means that sub-TLVs are present. If length is less than 8, the TLV MUST be ignored.

Zone ID (6 bytes): It is the identifier (ID) of a zone.

Flags field (16 bits): one flag bit E, OP of 3 bits, and a reserved subfield are as follows:

RESV: Reserved. MUST be send as zero and ignored on receipt.
E = 1: Indicating a node is a zone edge node
E = 0: Indicating a node is a zone internal node

When a Zone ID is configured on a zone node (refer to Section 6.1), the node updates its LSP by adding an IS-IS Zone ID TLV with the Zone ID. If it is a zone internal node, the TLV has its flag E = 0; otherwise (i.e., it is a zone edge node) the TLV has its flag E = 1 and includes a Zone ISN Sub TLV containing the zone links configured. Every link of a zone internal node is a zone link.

OP Value	Meaning (Operation)
0x001 (T):	Advertising Zone Topology Information for Migration
0x010 (M):	Migrating Zone to a Virtual Entity such as Virtual Node
0x011 (N):	Advertising Normal Topology Information for Rollback
0x100 (R):	Rolling Back from the Virtual Entity

The value of OP indicates one of the four operations above. When any of the other values is received, the TLV MUST be ignored.

The first two values of OP (i.e., OP = 0x001 and OP = 0x010) are used for transforming a zone to a TTZ virtual entity (refer to Section 5.1). The last two values (i.e., OP = 0x011 and OP = 0x100) are used for transforming (or say rolling back) the TTZ virtual entity back to the zone (refer to Section 5.2).

Two new sub-TLVs are defined, which may be added to an IS-IS Zone ID TLV. One is the Zone IS Neighbor sub-TLV, or Zone ISN sub-TLV for short. The other is the Zone ES Neighbor sub-TLV, or Zone ESN sub-TLV for short. A Zone ISN sub-TLV contains the information about a number of IS neighbors in the zone connected to a zone edge node. It has the format below.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Type (1)    |    Length     |        Neighbor ID(i)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               +
|                                                               |
+               +-----------------------------------------------+
|               |          Metric (i)                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                                                               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 4: Zone ISN Sub TLV

A Zone ISN Sub TLV has 1 byte of Type, 1 byte of Length of n*(IDLength + 3), which is followed by n tuples of Neighbor ID and Metric.

A Zone ESN sub-TLV contains the information about a number of ES neighbors in the zone connected to a zone edge node. It has the format below.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Type (2)    |    Length     |        Neighbor ID(i)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               +
|                                                               |
+               +-----------------------------------------------+
|               |          Metric (i)                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                                                               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 5: Zone ESN Sub TLV

After a zone ID is configured on a zone internal node (refer to Section 6.1), the zone internal node includes a Zone ID TLV with the zone ID and E = 0 in its LSP. The TLV indicates that the node originates the TLV is a zone internal node and all its links are zone links.

After a zone ID is configured on every zone link of a zone edge/border node (refer to Section 6.1), the zone edge/border node includes a Zone ID TLV with the zone ID, E = 1, Zone ISN Sub TLV and Zone ESN Sub TLV in its LSP. The TLV indicates that the node originates the TLV is a zone edge/border node and all the links in the Sub TLVs are zone links.

After all the zone nodes in a zone include their Zone ID TLVs in their LSPs, the zone is determined from the point of view of LSDB.

4.3. Zone as Edges Full Mesh

4.3.1. An Example of Zone as Edges Full Mesh

The figure below illustrates an area from the point of view on a router outside of TTZ 600 after TTZ 600 is created and abstracted as its edges full mesh from Figure 1.

```
             TTZ 600
               \
                \ ^~^~^~^~^~^~^~^~^~^~^
                 (                     )
===[R15]========(==[R61]=========[R63]==)======[R29]===
    ||\        (   ||  \\      //   ||   )       ||
    || \       (   ||   \\    //    ||   )       ||
    ||  \      (   ||    \\  //     ||   )       ||
    ||   \_____ (   ||     \\//      ||   )       ||
    ||        \(   ||      //\      ||   )       ||
    ||         \   ||     // \\     ||   )       ||
    ||         (\  ||    //   \\    ||   )       ||
    ||         ( \ ||   //     \\   ||   )       ||
    ||         (  \||  //       \\  ||   )       ||
===[R17]========(==[R65]=========[R67]==)=======[R31]===
     \\          (//                \\)        //
      ||         //v~v~v~v~v~v~v~v~v~v~v\\       ||
      ||        //                      \\     ||
      ||       //                        \\    ||
       \\     //                          \\  //
   ======[R23]============================[R25]=====
         //                                   \\
        //                                     \\
```

Figure 6: TTZ 600 as Edges Full Mesh

From a router outside of the TTZ, a TTZ is seen as the TTZ edge routers connected each other. For instance, router R15 sees that R61, R63, R65 and R67 are connected each other. The cost from one edge router to another edge router is the cost of the shortest path between these two routers.

The adjacencies between each of the TTZ's edge routers and its neighbors outside the TTZ are not changed. A router outside of the TTZ sees TTZ edge routers having their normal original connections to the routers outside of the TTZ. For example, router R15 sees that R61, R63, R65 and R67 have their normal original connections to R15, R29, R17 and R23, R25 and R31 respectively.

4.3.2. Updating LSPs for Zone as Edges Full Mesh

For every zone edge node, it updates its LSP in three steps and floods the LSP to all its neighbors.

At first, it adds each of the other zone edge nodes as an IS neighbor into the Intermediate System Neighbors TLV in its LSP after it receives an LSP containing an IS-IS Zone ID TLV with OP = M or a command activating migration zone to a TTZ virtual entity. The metric to the neighbor is the metric of the shortest path to the edge node within the zone.

In addition, it adds an IP internal reachability TLV into its LSP. The TLV contains a number of IP prefixes in the zone to be reachable from outside of the zone.

And then it removes the IS neighbors corresponding to the IS neighbors in the Zone ID TLV (i.e., in the Zone ISN sub TLV) from Intermediate System Neighbors TLV in its LSP, and the ES neighbors corresponding to the ES neighbors in the Zone ID TLV (i.e., in the Zone ESN sub TLV) from End System Neighbors TLV in the LSP. This SHOULD be done after it receives the LSPs for virtualizing zone from the other zone edges for a given time.

4.3.3. Computation of Routes

After a zone is transferred/migrated to the zone edges' full mesh, every zone node computes the routes (i.e., shortest paths to the destinations) using the graph consisting of the zone topology and the topology outside of the zone without the full mesh. Every node outside the zone computes the routes using the topology outside of the zone with the full mesh. The metric of every link inside and outside of the zone is not changed.

4.4. Advertisement of LSPs

LSPs can be divided into a couple of classes according to their Advertisements. The first class of LSPs is advertised within a zone. The second is advertised through a zone.

4.4.1. Advertisement of LSPs within Zone

Any LSP about a link state in a zone is advertised only within the zone. It is not advertised to any router outside of the zone. For example, a LSP generated for a zone internal node is advertised only within the zone.

Any LSP generated for a broadcast network in a zone is advertised only within the zone. It is not advertised outside of the zone.

After migrating to a zone as a single virtual node or edges' full mesh, every zone edge MUST NOT advertise any LSP belonging to the zone or any information in a LSP belonging to the zone to any node outside of the zone. The zone edge determines whether an LSP is about a zone internal link state by checking if the originating node of the LSP is a zone internal node.

For any LSP originated by a node within the zone, every zone edge node MUST NOT advertise it to any node outside of the zone.

4.4.2. Advertisement of LSPs through Zone

Any LSP about a link state outside of a zone received by a zone edge is advertised using the zone as transit. For example, when a zone edge node receives an LSP from a node outside of the zone, it floods the LSP to its neighbors both inside and outside of the zone.

The nodes in the zone continue to flood the LSP. When another zone edge receives the LSP, it floods the LSP to its neighbors both inside and outside of the zone.

5. Seamless Migration

This section presents the seamless migration between a zone and its single virtual node.

5.1. Transfer Zone to a Single Node

Transferring a zone to a single virtual node smoothly takes a few steps or stages.

At first, a user configures the zone on every node of the zone (refer to Section 6.1). Every zone node updates its LSP by including a Zone ID TLV. For a zone edge node, the TLV has the Zone ID configured, its flag E = 1 and a Zone ISN Sub TLV containing the zone links configured. For a zone internal node, the TLV has the Zone ID configured and its flag E = 0.

Second, after finishing the configuration of the zone, a user may issue a command, such as a CLI command, on a zone node, such as the zone leader, to trigger transferring the zone to the single virtual node. When the node receives the command, it updates its LSP by setting OP = T in its Zone ID TLV, which is distributed to every zone node. After receiving the Zone ID TLV with OP = T, every zone edge node, acting as a proxy of the virtual node, establishes a new adjacency between the virtual node and each of its zone neighbor nodes.

The command may be replaced by the determination made by a zone node, such as the zone leader. After determining that the configuration of the zone is finished for a given time such as 10 seconds, it updates its LSP by setting OP = T in its Zone ID TLV. The configuration is complete if every zone link configured is bidirectional. For every zone internal node configured with the Zone ID, there is an LSP containing its Zone ID TLV with E = 0 in the LSDB, which indicates that each link from the node (one direction) is a zone link. For every zone edge node, each of its zone links configured from the edge node (one direction) is included in its LSP containing its Zone ID TLV with E = 1 and Zone ISN Sub TLV in the LSDB.

Third, after receiving the updated LSPs from all the zone neighbor nodes, the zone leader checks if all the new adjacencies between the virtual node and the zone neighbor nodes have been established. If so, it originates an LS for the virtual node and updates its LSP (i.e., the LSP for itself zone leader) by setting OP = M in its Zone ID TLV, which is distributed to every zone node.

After receiving the LS for the virtual node or the Zone ID TLV with OP = M, every zone node migrates to zone as virtual node. Every zone edge node does not send any LS inside the zone to any zone neighbors. It advertises its LSP without any zone links to the nodes outside of the zone or purges its LSP outside of the zone, terminates its adjacency to each of its zone neighbors, but contains the adjacency in its LSP that is distributed within the zone. Every zone node computes the routes according to Section 4.1.5.

5.2. Roll Back from Zone as a Single Node

After abstracting a zone to a single virtual node, we may want to roll back the node to the zone smoothly in some cases. The process of rolling back has a few steps or stages.

At first, a user issues a command, such as a CLI command, on a zone node, such as the zone leader, to start (or prepare) for roll back. When receiving the command, the node triggers the preparation for roll back through updating its LSP by setting OP = N in its Zone ID TLV, which will be distributed to every node in the zone. After receiving the Zone ID TLV with OP = N, every zone edge node establishes a normal adjacency between the edge node and each of its zone neighbor nodes, and advertises the link state of the zone over the adjacency if it crosses the adjacency, but holds off its LSP containing the normal adjacency.

Second, a user may issue a command, such as a CLI command, on a zone node, such as the zone leader, to roll back from the virtual node to the zone if the following conditions are met.

Condition 1: All the normal adjacencies between every zone edge node and each of its zone neighbor nodes have been established.

Condition 2: All the link state about the zone that is supposed to be advertised outside of the zone has been advertised.

After receiving the command, the node updates its LSP by setting OP = R in its Zone ID TLV, which is distributed to every zone node. After receiving the Zone ID TLV with OP = R,

* every zone edge node, acting as a proxy of the virtual node, terminates the adjacency between the virtual node and each of its zone neighbor nodes and advertises its LSP containing the normal adjacencies between it and each of its zone neighbor nodes;

* The zone leader purges the LS for the virtual node abstracted from the zone; and

* Every zone node rolls back to normal.

The command may be replaced by the determination made by a zone node, such as the zone leader. After determining that all the conditions are met, it updates its LSP by setting OP = R in its Zone ID TLV, which is distributed to every zone node.

Condition 1 is met if it has its LSDB containing the link from each zone neighbor node to its zone edge node. That is that for every link from a zone neighbor node to the virtual node in the LSDB, there is a corresponding link from the zone neighbor to a zone edge node.

Condition 2 is met after Condition 1 has been met for a given time, such as maximum LSP advertisement time (MaxLSPAdvTime) crossing a network. We may assume that MaxLSPAdvTime is 5 seconds.

6. Operations

6.1. Configuring Zone

In general, a zone is a subset of an area and has a zone ID. It consists of some zone internal nodes and zone edge nodes. To configure it, a user configures this zone ID on every zone internal node and on every zone link of each zone edge node. A zone ID MUST be unique in an AS. It MUST NOT be any IP address in the AS from which a system ID is transformed to and used.

When the configuration of the zone ID is not consistent across the zone, some unexpected results will be generated. For example, when two different zone IDs are configured for the zone, two virtual nodes for two zones may be seen in the network. These are not expected. Once the unexpected results are seen, the inconsistent configurations MUST be fixed.

A node configured with the zone ID has all its links to be the zone links. The zone internal nodes and all their links plus the zone edge nodes and their zone links constitute the zone.

In a special case, a zone is an entire area and has a zone ID. All the links in the area are the zone links of the zone. To configure this zone, a user configures the zone ID on every zone node.

6.2. Transferring Zone to Node

Transferring a zone to a single virtual node smoothly may take a few steps or stages.

At first, a user configures the zone on every node of the zone.

After finishing the configuration of the zone, the user may issue a command, such as a CLI command, on a zone node, such as the zone leader, to trigger transferring the zone to the node (refer to Section 5.1).

If automatic transferring zone to node is enabled, the user does not need to issue the command. A zone node, such as the zone leader, will trigger transferring the zone to the node after determining that the configuration of the zone has been finished.

Then, all the zone nodes, including the zone leader, zone edge nodes and zone internal nodes, work together to make the zone to appear as a single virtual node smoothly in a couple of steps.

6.3. Rolling back Node to Zone

After abstracting a zone to a single virtual node, we may want to roll back the node to the zone smoothly in some cases. The process of rolling back has a few steps or stages.

At first, a user issues a command, such as a CLI command, on a zone node, such as the zone leader, to start (or prepare) for roll back. When receiving the command, the node triggers the preparation for roll back (refer to Section 5.2).

Second, a user may issue a command, such as a CLI command, on a zone node, such as the zone leader, to roll back from the virtual node to the zone if it is ready for roll back (refer to Section 5.2).

If automatic roll back Node to Zone is enabled, the user does not need to issue the command. A zone node, such as the zone leader, will trigger the roll back after determining that it is ready for roll back.

7. Experiment Scope

The experiment on TTZ should focus on node model. The experiment on TTZ mesh model in OSPF has been done. The experiment includes the aspects as follows.

* Abstraction. A zone (i.e., a block of an area not using TTZ) is abstracted as a single virtual node. The size of the LSDB for the area is reduced. Every node outside of the zone will see the virtual node and the other nodes outside of the zone after the abstraction. It will not see any node in the zone including the edge nodes of the zone.

* Separation. Any node that is not participating in a zone does not need to know or support TTZ.

* Safety. When a zone is configured correctly, neither zone edge node or zone internal node breaches after the zone is abstracted as a single virtual node.

* Alarm on Misconfiguration. Some critical misconfigurations should be detected and alarmed.

8. Security Considerations

The mechanism described in this document does not raise any new security issues for the IS-IS protocols. It is possible that an attacker may become or act as a zone leader and inject bad LSPs for the zone into the network, which disturbs the operations on the network, especially the IS-IS protocols. Authentication methods described in [RFC5304] and [RFC5310] SHOULD be used to prevent such attack.

9. IANA Considerations

IANA is requested to make a new allocation in the "IS-IS TLV Codepoint Registry" under the registry name "IS-IS TLV Codepoints" as follows:

TLV Type	TLV Name	reference
TBD1	Zone ID	This document

Note that TBD1 is less than 255.

IANA is requested to create a new sub-registry "Sub-TLVs for TLV type TBD1 (Zone ID TLV)" on the IANA IS-IS TLV Codepoints web page as follows:

Type	Name	reference
0	Reserved	
1	Zone ISN	This document
2	Zone ESN	This document
3 - 255	Unassigned	

10. References

10.1. Normative References

[ISO10589] ISO, "Intermediate System to Intermediate System Intra-Domain Routing Exchange Protocol for use in Conjunction with the Protocol for Providing the Connectionless-mode Network Service (ISO 8473)", ISO/IEC 10589:2002, November 2002.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/

RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[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>.

[RFC5304] Li, T. and R. Atkinson, "IS-IS Cryptographic Authentication", RFC 5304, DOI 10.17487/RFC5304, October 2008, <https://www.rfc-editor.org/info/rfc5304>.

[RFC5310] Bhatia, M., Manral, V., Li, T., Atkinson, R., White, R., and M. Fanto, "IS-IS Generic Cryptographic Authentication", RFC 5310, DOI 10.17487/RFC5310, February 2009, <https://www.rfc-editor.org/info/rfc5310>.

[I-D.ietf-lsr-dynamic-flooding] Li, T., Psenak, P., Chen, H., Jalil, L., and S. Dontula, "Dynamic Flooding on Dense Graphs", Work in Progress, Internet-Draft, draft-ietf-lsr-dynamic-flooding-17, 16 March 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-lsr-dynamic-flooding-17>.

10.2. Informative References

[RFC8099] Chen, H., Li, R., Retana, A., Yang, Y., and Z. Liu, "OSPF Topology-Transparent Zone", RFC 8099, DOI 10.17487/RFC8099, February 2017, <https://www.rfc-editor.org/info/rfc8099>.

Acknowledgments

The authors would like to thank Acee Lindem, Adrian Farrel, Abhay Roy, Christian Hopps, Dean Cheng, Russ White, Tony Przygienda, Wenhu Lu, Lin Han, Donald Eastlake, Tony Li, Robert Raszuk, Padmadevi Pillay Esnault, and Yang Yu for their valuable comments on TTZ.

Contributors

Alvaro Retana
Futurewei
Raleigh, NC
USA

Email: alvaro.retana@futurewei.com

Authors' Addresses

Huaimo Chen
Futurewei
Boston, MA
United States of America

Email: hchen.ietf@gmail.com

Richard Li
Futurewei
2330 Central expressway
Santa Clara, CA
United States of America

Email: richard.li@futurewei.com

Yi Yang
IBM
Cary, NC
United States of America

Email: yyietf@gmail.com

Anil Kumar S N
RtBrick
Bangalore
India

Email: anil.ietf@gmail.com

Yanhe Fan
Casa Systems
United States of America

Email: yfan@casa-systems.com

Ning So
Plano, TX 75082
United States of America

Email: ningso01@gmail.com

Vic Liu
United States of America

Email: liu.cmri@gmail.com

Mehmet Toy
Verizon

United States of America

Email: mehmet.toy@verizon.com

Lei Liu
Fujitsu
United States of America

Email: liulei.kddi@gmail.com

Kiran Makhijani
Futurewei
United States of America

Email: kiranm@futurewei.com