INTERNET-DRAFT
Updates: 6325, 6361, 7173
Intended status: Proposed Standard
Expires: September 8, 2015

Donald Eastlake
Huawei
Dacheng Zhang
Alibaba
March 9, 2015

TRILL: Link Security
<draft-eastlake-trill-link-security-00.txt>

Abstract

The TRILL protocol supports arbitrary link technologies between TRILL switches, both point-to-point and broadcast links, and supports Ethernet links between edge TRILL switches and end stations. Communications links are constantly under attack by criminals and national intelligence agencies as discussed in RFC 7258. Link security is an important element of security in depth, particularly for links that are not entirely under the physical control of the TRILL network operator or that include device which may have been compromised. This document specifies link security recommendations for TRILL over Ethernet, PPP, and pseudowire links taking into account performance considerations. It updates RFC 6325, 6361, and 7173. It requires that all TRILL packets between links ports capable of encryption at line speed MUST default to being encrypted. [This is an early partial draft.]

Table of Contents

1. Introduction

[This is an early partial draft.]

The TRILL (Transparent Interconnection of Lots of Links or Tuneled Routing in the Link Layer) protocol supports arbitrary link technologies including both point-to-point and broadcast links and supports Ethernet links between edge TRILL switches and end stations. Communications links are constantly under attack by criminals and national intelligence agencies as discussed in [RFC7258]. Link security in an important element of security in depth for links, paticularly those that are not entirely under the physical control of the TRILL network operator or that include device which may have been compromised.

TRILL generally uses an existing link security method specified for the technology of the link in question. TRILL provides autoconfiguration assistance and default keying material, under most circumstances, to support the TRILL goal of having a minimal or zero configuration default. Where better security is not available, TRILL supports opportunistic security [RFC7435].

This document specifies security recommendations for TRILL over Ethernet [RFC6325], TRILL over PPP [RFC6361], and transport of TRILL by pseudowires [RFC7173], in Sections 3, 4, and 5 respectively. Although the Security Considerations sections of these RFCs mention link security, this document goes further, updating these RFCs as decribed in Appendix A and imposing the new encryption requirement summarized in Section 1.1.

[TRILL-IP] is expected to cover TRILL security over IP links.

1.1 Encryption Requirement and Adjacency

This document requires that all TRILL packets between TRILL switch ports that are capable of encryption at line speed MUST default to being encrypted and authenticated. It MUST require explicit configuration in such cases for the ports to communicate unencrypted or unsecured. Line speed encrption and authentication usually requires hardware assist but there are cases with slower ports and higher powered switch processors where it can be accomplished in sofware.

If line speed encryption and authentication is not available for communication between TRILL switch ports, it MUST still be possible to configure the TRILL switches and ports involved to encrypt and authenticate all TRILL packets sent for cases where the security provided outweighs any reduction in performance.

1.2 Terminology and Acronyms

This document uses the acronyms and terms defined in [RFC6325], some of which are repeated below for convenience, and additional acronyms and terms listed below.

HKDF: Hash based Key Derivation Function [RFC5869].

Link: The means by which adjacent TRILL switches are connected. May be various technologies and in the common case of Ethernet, can be a "bridged LAN", that is to say, some combination of Ethernet links with zero or more bridges, hubs, repeaters, or the like.

MACSEC: Media Access Control (MAC) Security. IEEE Std 802.1AE-2006.

MPLS: Multi-Protocol Label Switching.

PPP: Point-to-point protocol [RFC1661].

RBridge: An alternative name for a TRILL switch.

TRILL: Transparent Interconnection of Lots of Links or Tunneled Routing in the Link Layer.

TRILL switch: A device implementing the TRILL protocol. An alternative name for an RBridge.

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].

2. Link Security Default Keying

In some cases, it is possible to use keying material derived from the [RFC5310] IS-IS keying material already in place. In such cases, the two byte [RFC5310] Key ID identifies the IS-IS keying material. The keying material actually used in the link security protocol is derived from the IS-IS keying material as follows:

```
HKDF-Expand-SHA256 ( IS-IS-key, "TRILL Link" | custom, L )
```

where "|" indicates concatenation, HKDF is the Hash base Key Derivation Function in [RFC5869], SHA256 is as in [RFC6234], IS-IS-key is the input keying material, "TRILL Link" is the 10-character ASCII [RFC20] string indicated, "custom" is a byte string dependeng on the link security protocol being used, and L is the length of output keying material needed.

3. Ethernet Links

TRILL over Ethernet is specified in [RFC6325] with some additional material on Ethernet link MTU in [rfc7180bis].

Link security between TRILL switch Ethernet ports conforms to IEEE Std 802.1AE-2006 [802.1AE] as amended by IEEE Std 802.1AEbn-2011 [802.1AEbn] and IEEE Std 802.1AEbw-2013 [802.1AEbw]. This security is referred to as MACSEC.

3.1 Between TRILL Switches

TRILL switch Ethernet ports MUST implement MACSEC. When TRILL switch ports are directly connected by Ethernet with no intervening customer bridges, for example by a point to point Ethernet link, MACSEC between them operates as specified herein. There can be intervening Provider Bridges or other forms of transparent Ethernet tunnels.

However, if there are one or more customer bridges or similar devices in the path, MACSEC at the TRILL switch port will peer with the nearest such bridge port. This reaults, from the point of view of MACSEC, with a two or more hop path. Typically, the TRILL switch ports at the ends of such a path would be unable to negotiate security and agree on keys so, in cases where encryption and authenication are required, they would be unable to establish IS-IS communication and would not form an adjacency [RFC7177]. However, it may be possible to configure such bridge ports and distribute such keying material or the like to them so that encryption and authentication can be established on all hops of such mulit-hop Ethernet paths. Methods for accomplishing such distribution to devices other than TRILL switches are beyond the scope of this document.

When MACSEC is established between adjacent TRILL switch ports, the frames are as shown in Figure 1. The optional VLAN tagging shown is superfluous in the case of TRILL Data and IS-IS packets. Unless there are VLAN sensitive devices intervening between the TRILL switch ports, or possibly attached to the link between those ports, TRILL Data and IS-IS packets SHOULD generally be sent untagged for efficiency.

Of course there may be other Ethernet control frames, such as link aggregation control messages or priority based flow control messages, that would also be sent within MACSEC. Typically only the [802.1X] messages used to establish and maintain MACSEC are sent unsecured.

```
                  +----------------------------------------+
                  |   Outer.MacDA (6 bytes)                |
                  +----------------------------------------+
                  |   Outer.MacSA (6 bytes)                |
                  +----------------------------------------+
                  |   MACSEC Tag (8 or 16 bytes)           |
                  +----------------------------------------+
                  | Encrypted                              |
                  |   +--------------------------------+   |
                  |   | Optional VLAN Tag (4 bytes)    |   |
                  |   +--------------------------------+   |
                  |   | TRILL or L2-IS-IS Ethertype    |   |
                  |   +--------------------------------+   |
                  |   | TRILL Data or IS-IS Payload    |   |
                  |   +--------------------------------+   |
                  +----------------------------------------+
                  |   ICV (8 or 16 bytes                   |
                  +----------------------------------------+
                  |   FCS (4 bytes)                        |
                  +----------------------------------------+
```

Figures 1. MACSEC Between TRILL Switch Ports

Outer.MacDA: 48-bit destination MAC address

Outer.MacSA: 48-bit source MAC address

MACSEC Tag: See further description below.

Encrypted: The encrypted data

ICV: The MACSEC Intergrity Check Value

FCS: Frame Check Sequence.

The strucutre of a MACSEC Tag is as follows:

tbd ...

3.1.1 Ethernet Link Security Maintenance

[802.1X] is used to establish keying and algorithms for Ethernet link security ... tbd ...

3.2 Ethernet Security to End Stations

MACSEC may be used between end stations and their adjacent TRILL switch(es) or end-to-end between end stations or both. Since TRILL does not impose administrative requirements on end stations, the choice of keying and crypto suite are beyond the scope of this document.

The end station must be properly configured to know if it should apply MACSEC to secure its connection to an edge TRILL switch or to remote end stations or both.

The Figure below show an Ethernet frame between a TRILL switch and the adjacent edge RBridge secured by MACSEC.

```
+---------------------------------------+
|  Outer.MacDA (6 bytes)                |
+---------------------------------------+
|  Outer.MacSA (6 bytes)                |
+---------------------------------------+
|  MACSEC Tag End Station to TRILL edge |
+---------------------------------------+
| Encrypted                             |
|    +-------------------------------+  |
|    | Optional VLAN Tag (4 bytes)   |  |
|    +-------------------------------+  |
|    | Payload Ethertype             |  |
|    +-------------------------------+  |
|    | Payload                       |  |
|    +-------------------------------+  |
+---------------------------------------+
|  ICV (8 or 16 bytes                   |
+---------------------------------------+
|  FCS (4 bytes)                        |
+---------------------------------------+
```

The Figure below shows an Ethernet frame between an end station and an adjacent edge RBridge where MACSEC is being used end-to-end between that end station and remote end stations.

```
+----------------------------------------+
|  Outer.MacDA (6 bytes)                 |
+----------------------------------------+
|  Outer.MacSA (6 bytes)                 |
+----------------------------------------+
|  Optional Outer.VLAN                   |
+----------------------------------------+
|  MACSEC Tag End Station to End Station|
+----------------------------------------+
| Encrypted                              |
|    +--------------------------------+  |
|    | Payload Ethertype              |  |
|    +--------------------------------+  |
|    | Payload                        |  |
|    +--------------------------------+  |
+----------------------------------------+
|  ICV (8 or 16 bytes                    |
+----------------------------------------+
|  FCS (4 bytes)                         |
+----------------------------------------+
```

The Figure below shows an Ethernet frame between an end station and an adjacent edge RBridge where MACSEC is being used end-to-end between that end station and remote end stations and, in addition, an outer application of MACSEC is securing traffic between the end station and the adjacent edge RBridge port.

```
+----------------------------------------------+
|  Outer.MacDA (6 bytes)                       |
+----------------------------------------------+
|  Outer.MacSA (6 bytes)                       |
+----------------------------------------------+
|  MACSEC Tag End Station to TRILL edge        |
+----------------------------------------------+
| Outer.Encrypted                              |
|   +-------------------------------------+    |
|   | Optional VLAN Tag (4 bytes)         |    |
|   +-------------------------------------+    |
|   | MACSEC Tag End Station to End Station|   |
|   +-------------------------------------+    |
|   | Inner.Encrypted                     |    |
|   |  +-------------------------------+  |    |
|   |  | Payload Ethertype             |  |    |
|   |  +-------------------------------+  |    |
|   |  | Payload                       |  |    |
|   |  +-------------------------------+  |    |
|   +-------------------------------------+    |
|   | Inner.ICV (8 or 16 bytes)           |    |
|   +-------------------------------------+    |
+----------------------------------------------+
|  Outer.ICV (8 or 16 bytes)                   |
+----------------------------------------------+
|  FCS (4 bytes)                               |
+----------------------------------------------+
```

4. PPP Links

TRILL over PPP is specified in [RFC6361]. Currently specified native PPP security does not meet modern security standards. However, true PPP over HDLC is relatively uncommon today and PPP is normally being conveyed by another protocol, such as PPP over Ethernet or PPP over IP. In those cases it is RECOMMENDED that Ethernet security as described in Section 3 or IP security as described in [TRILL-IP] be used to secure PPP between TRILL switch ports.

If it is necessary to use native PPP security [RFC1968] [RFC1994] ...tbd...

5. Pseudowire Links

TRILL transport over pseudowires is specified in [RFC7173].

No native security is provided for pseudowires as such; however, they are, by definition, carried by some PSN (Packet Switched Network). Link security must be provided by this PSN or by lower level protocols. This PSN is typically an MPLS or IP PSN.

In the case of a pseudowire over IP, security SHOULD be provided as is expected to be specified in [TRILL-IP]. If that is not possible but the IP path is only one IP hop, then it may be possible to provide link security at the layer of the link protocol supporting that hop, such as Ethernet (Section 3) or PPP (Section 4).

In the case of a pseudowire over MPLS, MPLS also does not have a native security scheme. Thus, security must be provided at the link layer being used, for example Ethernet (Section 3) or IP [TRILL-IP].

6. Security Considerations

This document is entirely about TRILL link security for Etherent, PPP, and pseudowire TRILL links. See sections of this document on those particular link technologies.

For general TRILL Security Considrations, see [RFC6325].

7. IANA Considerations

This document requires no IANA actions.

Normative References

[802.1AE] - IEEE Std 802.1AE-2006, IEEE Standard for Local and metropolitan networks / Media Access Control (MAC) Security, 18 August 2006.

[802.1AEbn] - IEEE Std 802.1AEbn-2011, IEEE Standard for Local and metropolitan networks / Media Access Control (MAC) Security / Galois Counter Mode - Advanced Encryption Standard - 256 (GCM-AES-256) Cipher Suite, 14 October 2011.

[802.1AEbw] - IEEE Std 802.1AEbw-2014, IEEE Standard for Local and metropolitan networks / Media Access Control (MAC) Security / Extended Packet Numbering, 12 February 2014

[RFC20] - Cerf, V., "ASCII format for network interchange", STD 80, RFC 20, October 1969, <http://www.rfc-editor.org/info/rfc20>.

[RFC1661] - Simpson, W., Ed., "The Point-to-Point Protocol (PPP)", STD 51, RFC 1661, July 1994, <http://www.rfc-editor.org/info/rfc1661>.

[RFC1968] - Meyer, G., "The PPP Encryption Control Protocol (ECP)", RFC 1968, June 1996, <http://www.rfc-editor.org/info/rfc1968>.

[RFC2119] -Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC5226] - T. Narten and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs," BCP 26 and RFC 5226, May 2008

[RFC5310] - Bhatia, M., Manral, V., Li, T., Atkinson, R., White, R., and M. Fanto, "IS-IS Generic Cryptographic Authentication", RFC 5310, February 2009.

[RFC5869] - Krawczyk, H. and P. Eronen, "HMAC-based Extract-and-Expand Key Derivation Function (HKDF)", RFC 5869, May 2010, <http://www.rfc-editor.org/info/rfc5869>

[RFC6234] - Eastlake 3rd, D. and T. Hansen, "US Secure Hash Algorithms (SHA and SHA-based HMAC and HKDF)", RFC 6234, May 2011, <http://www.rfc-editor.org/info/rfc6234>.

[RFC6325] - Perlman, R., Eastlake 3rd, D., Dutt, D., Gai, S., and A. Ghanwani, "Routing Bridges (RBridges): Base Protocol Specification", RFC 6325, July 2011, <http://www.rfc-editor.org/info/rfc6325>.

[RFC6361] - Carlson, J. and D. Eastlake 3rd, "PPP Transparent Interconnection of Lots of Links (TRILL) Protocol Control Protocol", RFC 6361, August 2011, <http://www.rfc-editor.org/info/rfc6361>.

[RFC7173] - Yong, L., Eastlake 3rd, D., Aldrin, S., and J. Hudson, "Transparent Interconnection of Lots of Links (TRILL) Transport Using Pseudowires", RFC 7173, May 2014, <http://www.rfc-editor.org/info/rfc7173>.

[RFC7177] = Eastlake 3rd, D., Perlman, R., Ghanwani, A., Yang, H., and V. Manral, "Transparent Interconnection of Lots of Links (TRILL): Adjacency", RFC 7177, May 2014, <http://www.rfc-editor.org/info/rfc7177>.

Informative References

[RFC1994] - Simpson, W., "PPP Challenge Handshake Authentication Protocol (CHAP)", RFC 1994, August 1996, <http://www.rfc-editor.org/info/rfc1994>.

[RFC2153] - W. Simpson, "PPP Vendor Extensions," RFC 2153, May 1997

[RFC3748] - B. Aboba, et al., "Extensible Authentication Protocol (EAP)," RFC 3748, June 2004

[RFC7258] - Farrell, S. and H. Tschofenig, "Pervasive Monitoring Is an Attack", BCP 188, RFC 7258, May 2014, <http://www.rfc-editor.org/info/rfc7258>.

[RFC7435] - Dukhovni, V., "Opportunistic Security: Some Protection Most of the Time", RFC 7435, December 2014, <http://www.rfc-editor.org/info/rfc7435>.

[rfc7180bis] - Eastlake, D., Zhang, M., Perlman, R. Banerjee, A., Ghanwani, A., and S. Gupta, "TRILL: Clarifications, Corrections, and Updates", draft-ietf-trill-rfc7180bis, work in progress.

[TRILL-IP] -

Acknowledgments

The authors thank the following for their comments and help:

tbd

Authors' Addresses

Donald Eastlake, 3rd
Huawei Technologies
155 Beaver Street
Milford, MA 01757 USA

Phone: +1-508-333-2270
Email: d3e3e3@gmail.com

Dacheng Zhang
Alibaba
Beijing, Chao yang District
P.R. China

Email: dacheng.zdc@alibaba-inc.com

Copyright and IPR Provisions

INTERNET-DRAFT Mingui Zhang
Intended Status: Proposed Standard Huawei
Radia Perlman
EMC
Hongjun Zhai
JIT
Muhammad Durrani
Brocade
Sujay Gupta
IP Infusion
Expires: August 9, 2015 February 5, 2015

TRILL Active-Active Edge Using Multiple MAC Attachments
draft-ietf-trill-aa-multi-attach-03.txt

Abstract

TRILL active-active service provides end stations with flow level load balance and resilience against link failures at the edge of TRILL campuses as described in RFC 7379.

This draft specifies a method by which member RBridges in an active-active edge RBridge group use their own nicknames as ingress RBridge nicknames to encapsulate frames from attached end systems. Thus, remote edge RBridges are required to keep multiple locations of one MAC address in one Data Label. Design goals of this specification are discussed in the document.

Table of Contents

1. Introduction

As discussed in [RFC7379], in a TRILL Active-Active Edge (AAE) topology, a Local Active-Active Link Protocol (LAALP), for example, a Multi-Chassis Link Aggregation Group (MC-LAG), is used to connect multiple RBridges to multi-port Customer Equipment (CE), such as a switch, vSwitch or a multi-port end station. An endnode clump is attached in the case of switch or vSwitch. It is required that data traffic within a specific VLAN from this endnode clump (including the multi-port end station case) can be ingressed and egressed by any of these RBridges simultaneously. End systems in the clump can spread their traffic among these edge RBridges at the flow level. When a link fails, end systems keep using the remaining links in the LAALP without waiting for the convergence of TRILL, which provides resilience to link failures.

Since a frame from each endnode can be ingressed by any RBridge in the AAE group, a remote edge RBridge may observe multiple attachment points (i.e., egress RBridges) for this endnode identified by its MAC address and Data Label (VLAN or Fine Grained Label (FGL)). This issue is known as the "MAC flip-flopping". Three potential solutions arise to address this issue:

1) AAE member RBridges use a pseudo-nickname, instead of their own, as the ingress nickname for end systems attached to the LAALP. [PN] falls within this category.

2) AAE member RBridges split work among themselves as to which one will be responsible for which MAC addresses. A member RBridge will encapsulate the frame using its own nickname if it is responsible for the source MAC address. Otherwise, if the frame is known unicast, it encapsulates the frame using the nickname of the responsible RBridge; if the frame is multi-destination, it needs to tunnel the native frame to its responsible RBridge for encapsulation, for example using [ChannelTunnel].

3) AAE member RBridges keep using their own nicknames. Remote edge RBridges are required to keep multiple points of attachment per MAC address and Data Label attached to the AAE.

The purpose of this document is to specify an approach based on solution 3. Although it focuses on exploring solution 3, the major design goals discussed here are common for all three AAE solutions. The use of any of these solutions in an AAE group does not prohibit the use of other solutions in other AAE groups in the same TRILL campus. For example, the specification in this draft and the specification in [PN] could be simultaneously deployed for different AAE groups in the same campus.

The main body of the document is organized as follows. Section 2 lists the acronyms and terminologies. Section 3 gives the overview model. Section 4 provides options for incremental deployment. Section 5 describes how this approach meets the design goals. The Sections after Section 5 cover security, IANA, and some backwards compatibility considerations.

2. Acronyms and Terminology

2.1. Acronyms and Terms

AAE: Active-Active Edge

Campus: a TRILL network consisting of TRILL switches, links, and possibly bridges bounded by end stations and IP routers. For TRILL, there is no "academic" implication in the name "campus"

CE : Customer Equipment (end station or bridge). The device can be either physical or virtual equipment.

Data Label: VLAN or FGL

DRNI: Distributed Resilient Network Interconnect. A link aggregation specified in [802.1AX] that can provide an LAALP between from 1 to 3 CEs and 2 or 3 RBridges.

Edge RBridge: An RBridge providing end station service on one or more of its ports.

ESADI: End Station Address Distribution Information [RFC7357]

FGL: Fine Grained Label [RFC7172]

IS-IS: Intermediate System to Intermediate System [ISIS]

LAALP: As in [RFC7379], Local Active-Active Link Protocol. Any protocol similar to MC-LAG (or DRNI) that runs in a distributed fashions on a CE, the links from that CE to a set of edge group RBridges, and on those RBridges.

MC-LAG: Multi-Chassis LAG. Proprietary extensions of Link Aggregation [802.1AX] that can provide an LAALP between one CE and 2 or more RBridges.

RBridge: A device implementing the TRILL protocol.

TRILL: TRansparent Interconnection of Lots of Links or Tunneled Routing in the Link Layer [RFC6325] [RFC7177].

TRILL switch: An alternative name for an RBridge.

vSwitch: A virtual switch such as a hypervisor that also simulates a bridge.

2.2. Terminology

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 RFC 2119 [RFC2119].

Familiarity with [RFC6325], [RFC6439] and [RFC7177] is assumed in this document.

3. Overview

```
                  +-----+
                  | RB4 |
       +----------+-----+----------+
       |                           |
       |                           |
       |       Rest of campus      |
       |                           |
       |                           |
       +-+-----+--+-----+--+-----+-+
         | RB1 |  | RB2 |  | RB3 |
         +-----\  +-----+  /-----+
                \    |    /
                  \  |  /
                   |||LAALP1
                   |||
                  +---+
                  | B |
                  +---+
                H1 H2 H3 H4: VLAN 10
```

Figure 3.1: An example topology for TRILL Active-Active Edge

Figure 3.1 shows an example network for TRILL Active-Active Edge. In this figure, endnodes (H1, H2, H3 and H4) are attached to a bridge B that communicates with multiple RBridges (RB1, RB2 and RB3) via the LAALP. Suppose RB4 is a 'remote' RBridge not in the AAE group in the TRILL campus. This connection model is also applicable to the virtualized environment where the physical bridge can be replaced with a vSwitch while those bare metal hosts are replaced with virtual machines (VM).

For a frame received from its attached endnode clumps, a member

RBridge of the AAE group conforming to this document always encapsulates that frame using its own nickname as the ingress nickname no matter whether it’s unicast or multicast.

The remote RBridge RB4 will see multiple attachments for each MAC from one of the end-nodes. Although this could cause problems if RB4 is learning remote end station attachments from the data plane, we specify a solution below ("Option C").

4. Incremental Deployable Options

Three options are listed below to handle incremental deployment scenarios. Among them, Option C can be incrementally implemented throughout a TRILL campus with common existing TRILL fast path hardware. Further details on Option C are given in Section 4.1.

-- Option A

A new capability announcement would appear in LSPs: "I can cope with data plane learning of multiple attachments for an endnode". This mode of operation is generally not supported by existing TRILL fast path hardware. Only if all edge RBridges to which the group has data connectivity and that are interested in any of the Data Labels in which the AAE is interested announce this capability can the AAE group safely use this approach. If all such RBridges do not announce this "Option A" capability, then a fallback would be needed such as reverting from active-active to active-standby operation or isolating the RBridge that would need to support this capability and do not support it. Further details for Options A are beyond the scope of this document except that in Section 4.2 a bit is reserved to indicate support for Option A because a remote RBridge supporting Option A is compatible with an AAE group using Option C.

-- Option B

Each edge RBridge in the AAE group ingresses frames from any LAALP into a specific TRILL topology [TRILL-MT]. In this way, the topology ID is used as the discriminator of different locations of a specific MAC address at the remote RBridge. TRILL could reserve a list of topology IDs to be dedicated to AAE. A variety of fallbacks might be needed for RBridges that do not support multi-topology or do not support a needed topology. Further details for this Options B are beyond the scope of this document.

-- Option C

As pointed out in Section 4.2.6 of [RFC6325] and Section 5.3 of

[RFC7357], one MAC address may be persistently claimed to be attached to multiple RBridges within the same Data Label in the TRILL ESADI-LSPs. For Option C, AAE member RBridges make use of the TRILL ESADI protocol to distribute multiple attachments of a MAC address. Remote RBridges SHOULD disable the data plane MAC learning for such multi-attached MAC addresses from TRILL Data packet decapsulation unless they also support Option A. The ability to configure an RBridge to disable data plane learning is provided by the base TRILL protocol [RFC6325].

4.1. Detail of Option C

With Option C, an RBridge in an AAE group MUST advertise all Data Labels enabled for all its attached LAALPs and participate in ESADI for those Data Labels. Receiver edge RBridges MUST avoid flip-flop errors in MAC learned from the TRILL Data packet decapsulation for the originating RBridge within these Data Labels. It’s RECOMMENDED that the receiver edge RBridge disable the data plane MAC learning from TRILL Data packet decapsulation within those advertised Data Labels for the originating RBridge unless the receiver RBridge also supports Option A. However, alternative implementations MAY be used to produce the same expected behavior. A promising way is to make use of the confidence level mechanism [RFC6325]. For example, let the receiver edge RBridge give a prevailing confidence value (e.g., 0x21) to the first MAC attachment learned from the data plane over others from the TRILL Data packet decapsulation. So the receiver edge RBridge will stick to this MAC attachment until it is overridden by one learned from the ESADI protocol [RFC7357]. The MAC attachment learned from ESADI is set to have higher confidence value (e.g., 0x80) to override any alternative learning from the decapsulation of received TRILL Data packets [RFC6325].

The advertisement of enabled Data Labels for an LAALP can be realized by allocating one reserved flag from the Interested VLANs and Spanning Tree Roots Sub-TLV (Section 2.3.6 of [RFC7176]) and one reserved flag from the Interested Labels and Spanning Tree Roots Sub-TLV (Section 2.3.8 of [RFC7176]). When this flag is set to 1, the originating IS (RBridge) is advertising Data Labels for LAALPs rather than plain LAN links. (See Section 8.3)

Whenever a MAC from the LAALP of this AAE is learned through ingress or configuration, it MUST be advertised via the ESADI protocol [RFC7357]. In its TRILL ESADI-LSPs, the originating RBridge needs to include the identifier of this AAE. Remote RBridges need to know all nicknames of RBridges in this AAE. This is achieved by listening to the "AA LAALP Group RBridges" TRILL APPsub-TLV defined in Section 5.3.2. The MAC Reachability TLVs [RFC6165] are composed in a way that each TLV only contains MAC addresses of end-nodes attached to a single LAALP. Each such TLV is enclosed in a TRILL APPsub-TLV defined as follows.

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Type = AA-LAALP-GROUP-MAC     | (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Length                        | (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| LAALP ID Size |                 (1 byte)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...+-+-+
| LAALP ID                        (k bytes)       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...+-+-+
| MAC-Reachability TLV            (7 + 6*n bytes) |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...+-+-+
```

o Type: AA LAALP Grouped MAC (TRILL APPsub-TLV type tbd1)

o Length: The MAC-Reachability TLV [RFC6165] is contained in the value field as a sub-TLV. The total number of bytes contained in the value field is given by k+8+6*n.

o LAALP ID Size: The length k of the LAALP ID in bytes.

o LAALP ID: The ID of the LAALP that is k bytes long. Here, it also serves as the identifier of the AAE. If the LAALP is an MC-LAG (or DRNI), it is the 8 byte ID as specified in Clause 6.3.2 in [802.1AX].

o MAC-Reachability sub-TLV: The AA-LAALP-GROUP-MAC APPsub-TLV value contains the MAC-Reachability TLV as a sub-TLV. As specified in Section 2.2 in [RFC7356], the type and length fields of the MAC-Reachability TLV are encoded as unsigned 16 bit integers. The one octet unsigned Confidence along with these TLVs SHOULD be set to prevail over those MAC addresses learned from TRILL Data decapsulation by remote edge RBridges.

This AA-LAALP-GROUP-MAC APPsub-TLV MUST be included in a TRILL GENINFO TLV [RFC7357] in the ESADI-LSP. There may be more than one occurrence of such TRILL APPsub-TLV in one ESADI-LSP fragment.

For those MAC addresses contained in an AA-LAALP-GROUP-MAC APPsub-TLV, this document applies. Otherwise, [RFC7357] applies. For example, an AAE member RBridge continues to enclose MAC addresses learned from TRILL Data packet decapsulation in MAC-Reachability TLV as per [RFC6165] and advertise them using the ESADI protocol.

When the remote RBridge learns MAC addresses contained in the AA-LAALP-GROUP-MAC APPsub-TLV via the ESADI protocol [RFC7357], it sends the packets destined to these MAC addresses to the closest one (the one to which the remote RBridge has the least cost forwarding path) of those RBridges in the AAE identified by the LAALP ID in the AA-LAALP-GROUP-MAC APPsub-TLV. If there are multiple equal least cost member RBridges, the ingress RBridge is required to select a unique one in a pseudo-random way as specified in Section 5.3 of [RFC7357].

When another RBridge in the same AAE group receives an ESADI-LSP with the AA-LAALP-GROUP-MAC APPsub-TLV, it also learns MAC addresses of those end-nodes served by the corresponding LAALP. These MAC addresses SHOULD be learned as if those end-nodes are locally attached to this RBridge itself.

An AAE member RBridge MUST use the AA-LAALP-GROUP-MAC APPsub-TLV to advertise in ESADI the MAC addresses learned from a plain local link (a non LAALP link) with Data Labels that happen to be covered by the Data Labels of any attached LAALP. The reason is that MAC learning from TRILL Data packet decapsulation within these Data Labels at the remote edge RBridge has normally been disabled for this RBridge.

4.2. Extended RBridge Capability Flags APPsub-TLV

The following Extended RBridge Capability Flags APPsub-TLV will be included in an E-L1FS FS-LSP fragment zero [RFC7180bis] as an APPsub-TLV of the TRILL GENINFO-TLV.

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Type = EXTENDED-RBRIDGE-CAP   | (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Length                        | (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Topology                      | (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|E|H|     Reserved                                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Reserved (continued)                                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- o Type: Extended RBridge Capability (TRILL APPsub-TLV type tbd2)

- o Length: Set to 8.

- o Topology: Indicates the topology to which the capabilities apply. When this field is set to zero, this implies that the capabilities apply to all topologies or topologies are not in use [TRILL-MT].

- o E: Bit 0 of the capability bits. When this bit is set, it indicates the originating IS acts as specified in Option C above.

o H: Bit 1 of the capability bits. When this bit is set, it indicates that the originating IS keeps multiple MAC attachments learned from TRILL Data packet decapsulation with fast path hardware, that is, it acts as specified in Option A above.

o Reserved: Flags extending from bit 2 through bit 63 of the capability fits reserved for future use. These MUST be sent as zero and ignored on receipt.

The Extended RBridge Capability Flags TRILL APPsub-TLV is used to notify other RBridges whether the originating IS supports the capability indicated by the E and H bits. For example, if E bit is set, it indicates the originating IS will act as defined in Option C. That is, it will disable the MAC learning from TRILL Data packet decapsulation within Data Labels advertised by AAE RBridges while waiting for the TRILL ESADI-LSPs to distribute the {MAC, Nickname, Data Label} association. Meanwhile, this RBridge is able to act as an AAE RBridge. It’s required to advertise MAC addresses learned from local LAALPs in TRILL ESADI-LSPs using the AA-LAALP-GROUP-MAC APPsub-TLV defined in Section 4.1. If the RBridge in an AAE group as specified herein observe a remote RBridge interested in one or more of that AAE group’s Data Labels and the remote RBridge does not support, as indicated by its extended capabilities, either Option A or Option C, then the AAE group MUST fall back to active-standby mode.

Capability specification for Option B is out the scope of this document.

5. Meeting the Design Goals

How this specification meets the major design goals of AAE is explored in this section.

5.1. No MAC Flip-Flopping (Normal Unicast Egress)

Since all RBridges talking with the AAE RBridges in the campus are able to see multiple locations for one MAC address in ESADI [RFC7357], a MAC address learned from one AAE member will not be overwritten by the same MAC address learned from another AAE member. Although multiple entries for this MAC address will be created, for return traffic the remote RBridge is required to adhere to a unique one of the locations (see Section 4.1) for each MAC address rather than keep flip-flopping among them.

5.2. Regular Unicast/Multicast Ingress

LAALP guarantees that each frame will be sent upward to the AAE via exactly one uplink. RBridges in the AAE can simply follow the process per [RFC6325] to ingress the frame. For example, each RBridge uses its own nickname as the ingress nickname to encapsulate the frame. In such a scenario, each RBridge takes for granted that it is the Appointed Forwarder for the VLANs enabled on the uplink of the LAALP.

5.3. Correct Multicast Egress

A fundamental design goal of AAE is that there must be no duplication or forwarding loop.

5.3.1. No Duplication (Single Exit Point)

When multi-destination TRILL Data packets for a specific Data Label are received from the campus, it’s important that exactly one RBridge out of the AAE group let through each multi-destination packet so no duplication will happen. The LAALP will have defined its selection function (using hashing or election algorithm) to designated a forwarder for a multi-destination frame. Since AAE member RBridges support the LAALP, they are able to utilize that selection function to determine the single exit point. If the output of the selection function points to the port attached to the receiver RBridge itself (i.e., the packet should be egressed out of this node), it MUST egress this packet for that AAE group. Otherwise, the packet MUST NOT be egressed for that AAE group. (It is output or not as specified in [RFC6325] updated by [RFC7172] for ports that lead to non-AAE links.)

5.3.2. No Echo (Split Horizon)

When a multi-destination frame originated from an LAALP is ingressed by an RBridge of an AAE group, distributed to the TRILL network and then received by another RBridge in the same AAE group, it is important that this RBridge does not egress this frame back to this LAALP. Otherwise, it will cause a forwarding loop (echo). The well known ’split horizon’ technique can be used to eliminate the echo issue.

RBridges in the AAE group need to split horizon based on the ingress RBridge nickname plus the VLAN of the TRILL Data packet. They need to set up per port filtering lists consists of the tuple of <ingress nickname, VLAN>. Packets with information matching with any entry of the filtering list MUST NOT be egressed out of that port. The information of such filters is obtained by listening to the following "LAALP Group RBridges" APPsub-TLV included in the TRILL GENINFO TLV in FS-LSPs [RFC7180bis].

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Type = AA-LAALP-GROUP-RBRIDGES| (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Length                        | (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Sender Nickname               | (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| LAALP ID Size |                 (1 byte)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...+-+-+
| LAALP ID                        (k bytes)       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...+-+-+
```

o Type: AA LAALP Grouped RBridges (TRILL APPsub-TLV type tbd3)

o Length: 3+k

o Sender Nickname: The nickname the originating IS will use as the ingress nickname. This field is useful because the originating IS might own multiple nicknames.

o LAALP ID Size: The length k of the LAALP ID in bytes.

o LAALP ID: The ID of the LAALP which is k bytes long. If the LAALP is an MC-LAG or DRNI, it is the 8-byte ID specified in Clause 6.3.2 in [802.1AX].

All enabled VLANs MUST be consistent on all ports connected to an LAALP. So the enabled VLANs need not be included in the AA-LAALP-GROUP-RBRIDGES TRILL APPsub-TLV. They can be locally obtained from the port attached to that LAALP.

Through parsing AA-LAALP-GROUP-RBRIDGES TRILL APPsub-TLVs, the receiver RBridge discovers all other RBridges connected to the same LAALP. The Sender Nickname of the originating IS will be added into the filtering list of the port attached to the LAALP. For example, RB3 in Figure 3.1 will set up a filtering list looks like {<RB1, VLAN10>, <RB2, VLAN10>} on its port attached to LAALP1. According to split horizon, TRILL Data packets within VLAN10 ingressed by RB1 or RB2 will not be egressed out of this port.

When there are multiple LAALPs connected to the same RBridge, these LAALPs may have overlap VLANs. Customer may need hosts within these overlap VLANs to communicate with each other. In Appendix A, several scenarios are given to explain how hosts communicate within the overlap VLANs and how split horizon happens.

5.4. No Black-hole or Triangular Forwarding

If a sub-link of the LAALP fails while remote RBridges continue to send packets towards the failed port, a black-hole happens. If the AAE member RBridge with that failed port starts to redirect the packets to other member RBridges for delivery, triangular forwarding occurs.

The member RBridge attached to the failed sub-link can make use of the ESADI protocol to flush those failure affected MAC addresses as defined in Section 5.2 of [RFC7357]. After doing that, no packets will be sent towards the failed port, hence no black-hole will happen. Nor will the member RBridge need to redirect packets to other member RBridges, which may otherwise lead to triangular forwarding.

5.5. Load Balance Towards the AAE

Since a remote RBridge can see multiple attachments of one MAC address in ESADI, this remote RBridge can choose to spread the traffic towards the AAE members on a per flow basis. Each of them is able to act as the egress point. In doing this, the forwarding paths need not be limited to the least cost Equal Cost Multiple Paths from the ingress RBridge to the AAE RBridges. The traffic load from the remote RBridge towards the AAE RBridges can be balanced based on a pseudo-random selection method (see Section 4.1).

Note that the load balance method adopted at a remote ingress RBridge is not to replace the load balance mechanism of LAALP. These two load spreading mechanisms should take effect separately.

5.6. Scalability

With option A, multiple attachments need to be recorded for a MAC address learned from AAE RBridges. More entries may be consumed in the MAC learning table. However, MAC addresses attached to an LAALP are usually only a small part of all MAC addresses in the whole TRILL campus. As a result, the extra space required by the multi-attached MAC addresses can usually be accommodated by RBridges unused MAC table space.

With option C, remote RBridges will keep the multiple attachments of a MAC address in the ESADI link state databases that are usually maintained by software. While in the MAC table that is normally implemented in hardware, an RBridge still establishes only one entry for each MAC address.

6. E-L1FS Backwards Compatibility

The Extended TLVs defined in Section 4 and 5 are to be used in an Extended Level 1 Flooding Scope (E-L1FS [RFC7356] [RFC7180bis]) PDU.

For those RBridges that do not support E-L1FS, the EXTENDED-RBRIDGE-CAP TRILL APPsub-TLV will not be sent out either and and MAC multi-attach active-active is not supported.

7. Security Considerations

Authenticity for contents transported in IS-IS PDUs is enforced using regular IS-IS security mechanism [ISIS][RFC5310].

For security considerations pertain to extensions transported by TRILL ESADI, see the Security Considerations section in [RFC7357].

For general TRILL security considerations, see [RFC6325].

8. IANA Considerations

8.1. TRILL APPsub-TLVs

IANA is requested to allocate three new types under the TRILL GENINFO TLV [RFC7357] for the TRILL APPsub-TLVs defined in Section 4.1, 4.2 and 5.3.2 of this document. The following entries are added to the "TRILL APPsub-TLV Types under IS-IS TLV 251 Application Identifier 1" Registry on the TRILL Parameters IANA web page.

Type	Name	Reference
tbd1[252]	AA-LAALP-GROUP-MAC	[This document]
tbd2[253]	EXTENDED-RBRIDGE-CAP	[This document]
tbd3[254]	AA-LAALP-GROUP-RBRIDGES	[This document]

8.2. Extended RBridge Capabilities Registry

IANA is requested to create a registry under the TRILL Parameters registry as follows:

Name: Extended RBridge Capabilities

Registration Procedure: Expert Review

Reference: [this document]

Bit	Mnemonic	Description	Reference
0	E	Option C Support	[this document]
1	H	Option A Support	[this document]
2-63	-	Unassigned	

8.3 Active Active Flags

IANA is requested to allocate two flag bits, with mnemonic "AA", as follows:

One flag bit appears in the "Interested VLANs and Spanning Tree Roots Sub-TLV".

Bit	Mnemonic	Description	Reference
0	M4	IPv4 Multicast Router Attached	[RFC7176]
1	M6	IPv6 Multicast Router Attached	[RFC7176]
2	-	Unassigned	
3	ES	ESADI Participation	[RFC7357]
4-15	-	(used for a VLAN ID)	[RFC7176]
16	AA	Enabled VLANs for Active-Active	[This document]
17-19	-	Unassigned	
20-31	-	(used for a VLAN ID)	[RFC7176]

One flag bit appears in the "Interested Labels and Spanning Tree Roots Sub-TLV".

Bit	Mnemonic	Description	Reference
0	M4	IPv4 Multicast Router Attached	[RFC7176]
1	M6	IPv6 Multicast Router Attached	[RFC7176]
2	BM	Bit Map	[RFC7176]
3	ES	ESADI Participation	[RFC7357]
4	AA	FGLs for Active-Active	[This document]
5-7	-	Unassigned	

9. Acknowledgements

Authors would like to thank the comments and suggestions from Andrew Qu, Donald Eastlake, Erik Nordmark, Fangwei Hu, Liang Xia, Weiguo Hao, Yizhou Li and Mukhtiar Shaikh.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC6165] Banerjee, A. and D. Ward, "Extensions to IS-IS for Layer-2 Systems", RFC 6165, April 2011.

[RFC6325] Perlman, R., Eastlake 3rd, D., Dutt, D., Gai, S., and A. Ghanwani, "Routing Bridges (RBridges): Base Protocol Specification", RFC 6325, July 2011.

[RFC6439] Perlman, R., Eastlake, D., Li, Y., Banerjee, A., and F. Hu, "Routing Bridges (RBridges): Appointed Forwarders", RFC 6439, November 2011.

[RFC7172] D. Eastlake 3rd and M. Zhang and P. Agarwal and R. Perlman and D. Dutt, "Transparent Interconnection of Lots of Links (TRILL): Fine-Grained Labeling", RFC 7172, May 2014.

[RFC7176] D. Eastlake 3rd and T. Senevirathne and A. Ghanwani and D. Dutt and A. Banerjee, "Transparent Interconnection of Lots of Links (TRILL) Use of IS-IS", RFC7176, May 2014.

[RFC7177] D. Eastlake 3rd and R. Perlman and A. Ghanwani and H. Yang and V. Manral, "Transparent Interconnection of Lots of Links (TRILL): Adjacency", RFC 7177, May 2014.

[RFC7356] Ginsberg, L., Previdi, S., and Y. Yang, "IS-IS Flooding Scope Link State PDUs (LSPs)", RFC 7356, September 2014.

[RFC7357] Zhai, H., Hu, F., Perlman, R., Eastlake 3rd, D., and O. Stokes, "Transparent Interconnection of Lots of Links (TRILL): End Station Address Distribution Information (ESADI) Protocol", RFC 7357, September 2014.

[RFC7379] Li, Y., Hao, W., Perlman, R., Hudson, J., and H. Zhai, "Problem Statement and Goals for Active-Active Connection at the Transparent Interconnection of Lots of Links (TRILL) Edge", RFC 7379, October 2014.

[RFC7180bis] D. Eastlake, M. Zhang, et al, "TRILL: Clarifications, Corrections, and Updates", draft-eastlake-trill-rfc7180bis, work in progress.

[802.1AX] IEEE, "IEEE Standard for Local and metropolitan area networks / Link Aggregation", 802.1AX-2014, 24 December 2014.

10.2. Informative References

[PN] H. Zhai, T. Senevirathne, et al, "TRILL: Pseudo-Nickname for Active-active Access", draft-ietf-trill-pseudonode-nickname, work in progress.

[ChannelTunnel] Eastlake, D. and Y. Li, "TRILL: RBridge Channel Tunnel Protocol", draft-ietf-trill-channel-tunnel, work in progress.

[TRILL-MT] D. Eastlake, M. Zhang, A. Banerjee, V. Manral, "TRILL:

Multi-Topology", draft-eastlake-trill-multi-topology, work in progress.

[ISIS] ISO, "Intermediate system to Intermediate system routeing information exchange protocol for use in conjunction with the Protocol for providing the Connectionless-mode Network Service (ISO 8473)", ISO/IEC 10589:2002.

[RFC5310] Bhatia, M., Manral, V., Li, T., Atkinson, R., White, R., and M. Fanto, "IS-IS Generic Cryptographic Authentication", RFC 5310, February 2009.

Appendix A. Scenarios for Split Horizon

```
+-------------------+   +-------------------+   +-------------------+
|        RB1        |   |        RB2        |   |        RB3        |
+-------------------+   +-------------------+   +-------------------+
L1       L2       L3   L1       L2       L3   L1       L2       L3
VL10˜20  VL15˜25  VL15 VL10˜20  VL15˜25  VL15 VL10˜20  VL15˜25  VL15
LAALP1   LAALP2   LAN  LAALP1   LAALP2   LAN  LAALP1   LAALP2   LAN
B1       B2       B10  B1       B2       B20  B1       B2       B30
```

Figure A.1: An example topology to explain split horizon

Suppose RB1, RB2 and RB3 are the Active-Active group connecting LAALP1 and LAALP2. LAALP1 and LAALP2 are connected to B1 and B2 at their other ends. Suppose all these RBridges use port L1 to connect LAALP1 while they use port L2 to connect LAALP2. Assume all three L1 enable VLAN 10˜20 while all three L2 enable VLAN 15˜25. So that there is an overlap of VLAN 15˜20. The customer needs hosts in these overlap VLANs to communicate with each other. That is, hosts attached to B1 in VLAN 15˜20 need to communicate with hosts attached to B2 in VLAN 15˜20. Assume the remote plain RBridge RB4 also has hosts attached in VLAN 15˜20 which need to communicate with those hosts in these VLANs attached to B1 and B2.

Two major requirements:

1. Frames ingressed from RB1-L1-VLAN 15˜20 MUST NOT be egressed out of ports RB2-L1 and RB3-L1. At the same time,

2. frames coming from B1-VLAN 15˜20 should reach B2-VLAN 15˜20.

RB3 stores the information for split horizon on its ports L1 and L2. On L1: {<ingress_nickname_RB1, VLAN 10˜20>, <ingress_nickname_RB2, VLAN 10˜20>} and on L2: {<ingress_nickname_RB1, VLAN 15˜25>, <ingress_nickname_RB2, VLAN 15˜25>}.

Five clarification scenarios:

a. Suppose RB2/RB3 receives a TRILL multi-destination data packet with VLAN 15 and ingress nickname RB1. RB3 is the single exit point (selected out according to the hashing function of LAALP) for this packet. On ports L1 and L2, RB3 has covered <ingress_nickname_RB1, VLAN 15>, so that RB3 will not egress this packet out of either L1 or L2. Here, _split horizon_ happens.

   Beforehand, RB1 obtains a native frame on port L1 from B1 in VLAN 15. RB1 judges it should be forwarded as a multi-destination packet across the TRILL campus. Also, RB1 replicates this frame without TRILL encapsulation and sends it out of port L2, so that B2 will get this frame.

b. Suppose RB2/RB3 receives a TRILL multi-destination data packet with VLAN 15 and ingress nickname RB4. RB3 is the single exit point. On ports L1 and L2, since RB3 has not stored any tuple with ingress_ nickname_RB4, RB3 will decapsulate the packet and egress it out of both ports L1 and L2. So both B1 and B2 will receive the frame.

c. Suppose there is a plain LAN link port L3 on RB1, RB2 and RB3, connecting to B10, B20 and B30 respectively. These L3 ports happen to be configured with VLAN 15. On port L3, RB2 and RB3 stores no information of split horizon for AAE (since this port has not been configured to be in any LAALP). They will egress the packet ingressed from RB1-L1 in VLAN 15.

d. If a packet is ingressed from RB1-L1 or RB1-L2 with VLAN 15, port RB1-L3 will not egress packets with ingress-nickname-RB1. RB1 needs to replicate this frame without encapsulation and sends it out of port L3. This kind of 'bounce' behavior for multi-destination frames is just as specified in paragraph 2 of Section 4.6.1.2 of [RFC6325].

e. If a packet is ingressed from RB1-L3, since RB1-L1 and RB1-L2 cannot egress packets with VLAN 15 and ingress-nickname-RB1, RB1 needs to replicate this frame without encapsulation and sends it out of port L1 and L2. (Also see paragraph 2 of Section 4.6.1.2 of [RFC6325].)

Author’s Addresses

Mingui Zhang
Huawei Technologies
No.156 Beiqing Rd. Haidian District,
Beijing 100095 P.R. China

EMail: zhangmingui@huawei.com

Radia Perlman
EMC
2010 256th Avenue NE, #200
Bellevue, WA 98007 USA

EMail: radia@alum.mit.edu

Hongjun Zhai
Jinling Institute of Technology
99 Hongjing Avenue, Jiangning District
Nanjing, Jiangsu 211169 China

EMail: honjun.zhai@tom.com

Muhammad Durrani
Brocade
130 Holger Way
San Jose, CA 95134

EMail: mdurrani@brocade.com

Sujay Gupta
IP Infusion,
RMZ Centennial
Mahadevapura Post
Bangalore - 560048
India

EMail: sujay.gupta@ipinfusion.com

W. Hao
Y. Li
Huawei Technologies
M. Durrani
Brocade
S. Gupta
IP Infusion
A. Qu
MediaTec
T. Han
Huawei Technologies

February 10, 2015

Centralized Replication for BUM traffic in active-active edge connection
draft-ietf-trill-centralized-replication-01.txt

Abstract

In TRILL active-active access scenario, RPF check failure issue may occur when pseudo-nickname mechanism in [TRILLPN] is used. This draft describes a solution to the RPF check failure issue through centralized replication for BUM (Broadcast, Unknown unicast, Mutlicast) traffic. The solution has all ingress RBs send BUM traffic to a centralized node via unicast TRILL encapsulation. When the centralized node receives the BUM traffic, it decapsulates the traffic and forwards the BUM traffic to all destination RBs using a distribution tree established via the TRILL base protocol. To avoid RPF check failure on a RBridge sitting between the ingress RBridge and the centralized replication node, some change of RPF calculation algorithm is required. RPF calculation on each RBridge should use the centralized node as ingress RB instead of the real ingress RBridge of RBv to perform the calculation.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

The IETF TRILL (Transparent Interconnection of Lots of Links) [RFC6325] protocol provides loop free and per hop based multipath data forwarding with minimum configuration. TRILL uses IS-IS [RFC6165] [RFC6326bis] as its control plane routing protocol and defines a TRILL specific header for user data.

Classic Ethernet device (CE) devices typically are multi-homed to multiple edge RBridges which form an edge group. All of the uplinks of CE are bundled as a Multi-Chassis Link Aggregation (MC-LAG). An active-active flow-based load sharing mechanism is normally implemented to achieve better load balancing and high reliability. A CE device can be a layer 3 end system by itself or a bridge switch through which layer 3 end systems access to TRILL campus.

In active-active access scenario, pseudo-nickname solution in [TRILLPN] can be used to avoid MAC flip-flop on remote RBs. The basic idea is to use a virtual RBridge of RBv with a single pseudo-nickname to represent an edge group that MC-LAG connects to. Any member RBridge of that edge group should use this pseudo-nickname rather than its own nickname as ingress nickname when it injects TRILL data frames to TRILL campus. The use of the nickname solves the address flip flop issue by making the MAC address learnt by the remote RBridge bound to pseudo-nickname. However, it introduces another issue, which is incorrect packet drop by RPF check failure. When a pseudo-nickname is used by an edge RBridge as the ingress nickname to forward BUM traffic, any RBridges sitting between the ingress RB and the distribution tree root will treat the traffic as it is ingressed from the virtual RBridge RBv. If same distribution tree is used by these different edge RBridges, the traffic may arrive at RBn from different ports. Then the RPF check fails, and some of the traffic receiving from unexpected ports will be dropped by RBn.

This document proposes a centralized replication solution for broadcast, unknown unicast, multicast(BUM) traffic to solve the issue of incorrect packet drop by RPF check failure. The basic idea is that all ingress RBs send BUM traffic to a centralized node which is recommended to be a distribution tree root using unicast TRILL encapsulation. When the centralized node receives that traffic, it decapsulates it and then forwards the BUM traffic to all destination RBs using a distribution tree established as per TRILL base protocol.

2. Conventions used in this document

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 RFC-2119 [RFC2119].The acronyms and terminology in [RFC6325] is used herein with the following additions:

BUM - Broadcast, Unknown unicast, and Multicast

CE - As in [CMT], Classic Ethernet device (end station or bridge).

The device can be either physical or virtual equipment.

3. Centralized Replication Solution Overview

When an edge RB receives BUM traffic from a CE device, it acts as ingress RB and uses unicast TRILL encapsulation instead of multicast TRILL encapsulation to send the traffic to a centralized node. The centralized node is recommended to be a distribution tree root.

The TRILL header of the unicast TRILL encapsulation contains an "ingress RBridge nickname" field and an "egress RBridge nickname" field. If ingress RB receives the traffic from the port which is in a MC-LAG, it should set the ingress RBridge nickname to be the pseudo-nickname rather than its own nickname to avoid MAC flip-flop on remote RBs as per [TRILLPN]. The egress RBridge nickname is set to the special nickname of the centralized node which is used to differentiate the unicast TRILL encapsulation BUM traffic from normal unicast TRILL traffic. The special nickname is called R-nickname.

When the centralized node receives the unicast TRILL encapsulated BUM traffic from ingress RB, the node decapsulates the packet. Then the centralized node replicates and forwards the BUM traffic to all destination RBs using one of the distribution trees established as per TRILL base protocol, if the centralized node is the root of a distribution tree, the recommended distribution tree is the tree whose root is the centralized node itself. When the centralized node forwards the BUM traffic, ingress nickname remains the same as that in frame it received to ensure that the MAC address learnt by all egress RBridges bound to pseudo-nickname.

When the replicated traffic is forwarded on each RBridge along the distribution tree starting from the centralized node, RPF check will be performed as per RFC6325. For any RBridge sitting between the ingress RBridge and the centralized replication node, the traffic incoming port should be the centralized node facing port as the multicast traffic always comes from the centralized node in this solution. However the RPF port as result of distribution tree calculation as per RFC 6325 will be the real ingress RB facing port as it uses virtual RBridge as ingress RB, so RPF check will fail. To solve this problem, some change of RPF calculation algorithm is required. RPF calculation on each RBridge should use the centralized node as ingress RB instead of the real ingress virtual RBridge to perform the calculation. As a result, RPF check will point to the centralized node facing port on the RBridge for multi-destination traffic. It prevents the incorrect frame discard by RPF check.

To differentiate the unicast TRILL encapsulation BUM traffic from normal unicast TRILL traffic on a centralized node, besides the centralized node’s own nickname, R-nickname should be introduced for centralized replication. Only when the centralized node receives unicast TRILL encapsulation traffic with egress nickname equivalent to the R-nickname, the node does unicast TRILL decapsulaton and then forwards the traffic to all destination RBs through a distribution tree. The centralized nodes should announce its R-nickname to all TRILL campus through TRILL LSP extension.

4. Frame duplication from remote RB

Frame duplication may occur when a remote host sends multi-destination frame to a local CE which has an active-active connection to the TRILL campus. To avoid local CE receiving multiple copies from a remote RBridge, the designated forwarder (DF) mechanism should be supported for egress direction multicast traffic.

DF election mechanism allows only one port in one RB of MC-LAG to forward multicast traffic from TRILL campus to local access side for each VLAN. The basic idea of DF is to elect one RBridge per VLAN from an edge group to be responsible for egressing the multicast traffic. [draft-hao-trill-dup-avoidance-active-active-02] describes the detail DF mechanism and TRILL protocol extension for DF election.

If DF-election mechanism is used for frame duplication prevention, access ports on an RB are categorized as three types: non mc-lag, mc-lag DF port and mc-lag non-DF port. The last two types can be called mc-lag port. For each of the mc-lag port, there is a pseudo-nickname associated. If consistent nickname allocation per edge group RBridges is used, it is possible that same pseudo-nickname associated to more than one port on a single RB. A typical scenario is that CE1 is connected to RB1 & RB2 by mc-lag1 while CE2 is connected to RB1 & RB2 by mc-lag 2. In order to save the number of pseudo-nickname used, member ports for both mc-lag1 and mc-lag2 on RB1 & RB2 are all associated to pseudo-nickname pn1.

5. Local forwarding behavior on ingress RBridge

When a ingress RBridge(RB1) receives BUM traffic from an active-active accessing CE(CE1) device, the traffic will be injected to TRILL campus through TRILL encapsulation, and it will be replicated and forwarded to all destination RBs which include ingress RB itself along a TRILL distribution tree. So the traffic will return to the ingress RBridge. To avoid the traffic looping back to original sender CE, ingress nickname can be used for traffic filtering.

If there are two local connecting CE(CE1 and CE2) devices on ingress RB, the BUM traffic between these two CEs can’t be forwarded locally and through TRILL campus simultaneously, otherwise duplicated traffic will be received by destination CE. Local forwarding behavior on ingress RBridge should be carefully designed.

To avoid duplicated traffic on receiver CE, local replication behavior on RB1 is as follows:

1. Local replication to the ports associated with the same pseudo-nickname as that associated to the incoming port.

2. Do not replicate to mc-lag port associated with different pseudo-nickname.

3. Do not replicate to non mc-lag ports.

The above local forwarding behavior on the ingress RB of RB1 can be called centralized local forwarding behavior A.

If ingress RB of RB1 itself is the centralized node, BUM traffic injected to TRILL campus won't loop back to RB1. In this case, the local forwarding behavior is called centralized local forwarding behavior B. The local replication behavior on RB1 is as follows:

1. Local replication to the ports associated with the same pseudo-nickname as that associated to the incoming port.

2. Local replication to the mc-lag DF port associated with different pseudo-nickname. Do not replicate to mc-lag non-DF port associated with different pseudo-nickname.

3. Local replication to non mc-lag ports.

6. Loop prevention among RBridges in a edge group

If a CE sends a broadcast, unknown unicast, or multicast (BUM) packet through DF port to a ingress RB, it will forward that packet to all or subset of the other RBs that only have non-DF ports for that MC-LAG. Because BUM traffic forwarding to non-DF port isn't allowed, in this case the frame won't loop back to the CE.

If a CE sends a BUM packet through non-DF port to a ingress RB, say RB1, then RB1 will forward that packet to other RBridges that have DF port for that MC-LAG. In this case the frame will loop back to the CE and traffic split-horizon filtering mechanism should be used to avoid looping back among RBridges in a edge group.

Split-horizon mechanism relies on ingress nickname to check if a packet's egress port belongs to a same MC-LAG with the packet's incoming port to TRILL campus.

When the ingress RBridge receives BUM traffic from an active-active accessing CE device, the traffic will be injected to TRILL campus through TRILL encapsulation, and it will be replicated and forwarded to all destination RBs which include ingress RB itself through TRILL distribution tree. If same pseudo-nickname is used for two active-active access CEs as ingress nickname, egress RB can use the nickname to filter traffic forwarding to all local CE. In this case, the traffic between these two CEs goes through local RB and another copy of the traffic from TRILL campus is filtered. If different ingress nickname is used for two connecting CE devices, the access ports connecting to these two CEs should be isolated with each other. The BUM traffic between these two CEs should go through TRILL campus, otherwise the destination CE connected to same RB with the sender CE will receive two copies of the traffic.

Do note that the above sections on techniques to avoid frame duplication, loop prevention is applicable assuming the Link aggregation technology in use is unaware of the frame duplication happening. For example using mechanisms like IEEE802.1AX, Distributed Resilient Network Interconnect (DRNI) specs implements mechanism similar to DF and also avoids some cases of frame duplication & looping.

7. Centralized replication forwarding process

```
                                   +-----------+
                                   |   (RB5)   |
                                   +-----------+
                                         |
                                   +-----------+
                                   |   (RB4)   |
                                   +-----------+

                                    |     |    |
                            --------      |    --------
                           |              |            |
                         +------+      +------+      +------+
                         |(RB1) |      |(RB2) |      | (RB3)|
                         +------+      +------+      +------+
                           *   |         *  |          * |  ^
                           *   |         *  |          * |   ^
                           *   ----------*-------------*--    ^
                           **************************** |      ^
                   MC-LAG1 *                     MC-LAG2 |       ^
                       +------+                     +------+    +------+
                       |  CE1 |                     | CE2  |    | CE3  |
                       +------+                     +------+    +------+
```

Figure 1 TRILL Active-active access

Assuming the centralized replication solution is used in the network of above figure 1, RB5 is the distribution tree root and centralized replication node, CE1 and CE2 are active-active accessed to RB1,RB2 and RB3 through MC-LAG1 and MC-LAG2 respectively, CE3 is single homed to RB3. The RBridge’s own nickname of RB1 to RB5 are nick1 to nick5 respectively. RB1,RB2 and RB3 use same pseudo-nickname for MC-LAG1 and MC-LAG2, the pseudo-nickname is P-nick. The R-nickname on the centralized replication node of RB5 is S-nick.

The BUM traffic forwarding process from CE1 to CE2,CE3 is as follows:

1. CE1 sends BUM traffic to RB3.

2. RB3 replicates and sends the BUM traffic to CE2 locally. RB2 also sends the traffic to RB5 through unicast TRILL encapsulation. Ingress nickname is set as P-nick, egress nickname is set as S-nick.

3. RB5 decapsulates the unicast TRILL packet. Then it uses the distribution tree whose root is RB5 to forward the packet. The egress nickname in the trill header is the nick5. Ingress nickname is still P-nick.

4. RB4 receives multicast TRILL traffic from RB5. Traffic incoming port is the up port facing to distribution tree root, RPF check will be correct based on the changed RPF port calculation algorithm in this document. After RPF check is performed, it forwards the traffic to all other egress RBs(RB1,RB2 and RB3).

5. RB3 receives multicast TRILL traffic from RB4. It decapsulates the multicast TRILL packet. Because ingress nickname of P-nick is equivalent to the nickname of local MC-LAGs connecting CE1 and CE2, it doesn’t forward the traffic to CE1 and CE2 to avoid duplicated frame. RB3 only forwards the packet to CE3.

6. RB1 and RB2 receive multicast TRILL traffic from RB4. The forwarding process is similar to the process on RB3, i.e, because ingress nickname of P-nick is equivalent to the nickname of local MC-LAGs connecting CE1 and CE2, they also don’t forward the traffic to local CE1 and CE2.

8. BUM traffic loadbalancing among multiple centralized nodes

To support unicast TRILL encapsulation BUM traffic load balancing, multiple centralized replication node can be deployed and the traffic can be load balanced on these nodes in vlan-based or flow-based mode.

8.1. Vlan-based loadbalancing

Assuming there are k centralized nodes in TRILL campus, each centralized node has different R-nickname, VLAN-based(or FGL-based, etc) loadbalancing algorithm used by ingress active-active access RBridge is as follows:

1. All centralized nodes are ordered and numbered from 0 to k-1

in ascending order according to the 7-octet IS-IS ID.

2. For VLAN ID m, choose the centralized node whose number equals (m mod k).

An example of the m mod K, is that for 3 centralized nodes (CN) and 5 VLANs is: VLAN 0 goes to CN0, VLAN1 goes to CN1, VLAN2 goes to CN2, VLAN4 goes to CN0, and VLAN5 goes to CN1.

When a ingress RBridge participating active-active connection receives BUM traffic from local CE, the RB decides to send the traffic to which centralized node based on the VLAN-based loadbalancing algorithm, vlan-based loadbalancing for the BUM traffic can be achieved among multiple centralized nodes.

8.2. Flow-based loadbalancing

To support flow-based loadbalancing for BUM traffic between different centralized node, anycast R-nickname mechanism should be introduced, which means a same R-nickname is attached to both physical centralized node at the same time. Each centralized node announces the R-nickname through the Nickname Sub-Tlv specified in [RFC6326] to TRILL network and MUST ignore the nickname collision check as defined in basic TRILL protocol.

The egress nickname of unicast TRILL encapsulation for BUM traffic from ingress RB is the R-nickname. The unicast TRILL encapsulation BUM traffic would go to any one of the physical centralized nodes by the natural support of equal cost multicast path (ECMP) from TRILL protocol.

The physical centralized node will decapsulate the unicast TRILL encapsulation and forward it through any one of the distribution trees established per RFC 6325 with the original source, and BUM destination. Because ECMP of the unicast TRILL encapsulation BUM traffic is supported among multiple centralized nodes, so it can achieve better link bandwidth usage than VLAN-based(or FGL-based, etc)loadbalancing.

9. Co-existing with CMT solution

```
                    +------+    +------+
                    |(RB6) |    |(RB7) |
                    +------+    +------+
      ------------------|-----------|----------------------
      |            |              |          |            |
   +------+    +------+       +------+    +------+     +------+
   |(RB1) |    |(RB2) |       |(RB3) |    |(RB4) |     |(RB5) |
   +------+    +------+       +------+    +------+     +------+
       |          |               |          |            |
       ------------               -------------------------
             |                               |
         +------+                         +------+
         |  CE1 |                         |  CE2 |
         +------+                         +------+
```

Figure 2 CMT and centralized replication co-existing scenario

Both the centralized replication solution and CMT solution rely on pseudo-nickname to avoid MAC flip-flop on remote RBridges, these two solutions can co-exist in one TRILL campus. Different edge group RBridges can select either the centralized replication solution or CMT solution independently to inject traffic to TRILL campus. As illustrated in figure 2, RB1 and RB2 use CMT for CE1’s active-active access, RB3,RB4 and RB5 use the centralized replication for CE2’s active-active access.

For the centralized replication solution, edge group RBridges should announce local pseudo-nickname using Nickname Flags APPsub-TLV with C-flag, the nickname with C-flag is called "C-nickname". A transit RBridge will perform different RPF check algorithm if it receives TRILL encapsulation traffic with C-nickname as ingress nickname.

10. Network Migration Analysis

Centralized nodes need software and hardware upgrade to support centralized replication process, which stitches TRILL unicast traffic decapsulation process and the process of normal TRILL multicast traffic forwarding along distribution tree.

Active-active connection edge RBs need software and hardware upgrade to support unicast TRILL encapsulation for BUM traffic, the process is similar to normal head-end replication process.

Transit nodes need software upgrade to support RPF port calculation algorithm change.

11. TRILL protocol extension

Two Flags of "R" and "C" in Nickname Flags APPsub-TLV [RFC7180bis] are introduced, the nickname with "R" flag is called R-nickname, the nickname with "C" flag is called C-nickname. R-nickname is set on one or multiple centralized nodes, R-nickname is a specialized nickname to differentiate unicast TRILL encapsulation BUM traffic from normal unicast TRILL traffic. C-nickname is set on edge group RBridges, C-nickname is a specialized pseudo-nickname for transit RBridges to perform different RPF check algorithm.

When active-active edge RBridges use centralized replication to forward BUM traffic, the R-nickname is used as the egress nickname and the C-nickname is used as ingress nickname in TRILL header for unicast TRILL encapsulation of BUM traffic.

11.1. "R" and "C" Flag in Nickname Flags APPsub-TLV

```
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
            |   Nickname                                    |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
            |IN|D |R | C|    RESV                           |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
                            NICKFLAG RECORD
```

o R. If R flag is one, it indicates that the advertising TRILL switch is a centralized replication node, and the nickname is used as egress nickname for edge group RBridges to inject traffic to TRILL campus when the edge group RBridges use centralized replication solution for active-active access. If flag is zero, that nickname will not be used for that purpose.

o C. If C flag is one, it indicates that the TRILL traffic with this nickname as ingress nickname requires special RPF check algorithm. If flag is zero, that nickname will not be used for that purpose.

12. Security Considerations

This draft does not introduce any extra security risks. For general TRILL Security Considerations, see [RFC6325].

13. IANA Considerations

This document requires no IANA Actions. RFC Editor: Please remove this section before publication.

14. References

14.1. Normative References

[1] [RFC6165] Banerjee, A. and D. Ward, "Extensions to IS-IS for Layer-2 Systems", RFC 6165, April 2011.

[2] [RFC6325] Perlman, R., et.al. "RBridge: Base Protocol Specification", RFC 6325, July 2011.

[3] [RFC6326bis] Eastlake, D., Banerjee, A., Dutt, D., Perlman, R., and A. Ghanwani, "TRILL Use of IS-IS", draft-eastlake-isis-rfc6326bis, work in progress.

[4] [RFC7180bis] Eastlake, D., Zhang, M., Perlman, R., Banerjee, A. Ghanwani and Gupta.S, "TRILL: Clarifications, Corrections, and Updates", draft-ietf-trill-rfc7180bis-00, work in progress.

14.2. Informative References

[1] [TRILLPN] Zhai,H., et.al., "RBridge: Pseduonode nickname", draft-hu-trill-pseudonode-nickname, Work in progress, November 2011.

[2] [TRILAA] Li,Y., et.al., " Problem Statement and Goals for Active-Active TRILL Edge", draft-ietf-trill-active-active-connection-prob-00, Work in progress, July 2013.

[3] [CMT] Senevirathne, T., Pathangi, J., and J. Hudson, "Coordinated Multicast Trees (CMT)for TRILL", draft-ietf-trill-cmt-00.txt Work in Progress, April 2012.

15. Acknowledgments

The authors wish to acknowledge the important contributions of Hongjun Zhai, Xiaomin Wu, Liang Xia.

Authors' Addresses

Weiguo Hao
Huawei Technologies
101 Software Avenue,
Nanjing 210012
China
Phone: +86-25-56623144
Email: haoweiguo@huawei.com

Yizhou Li
Huawei Technologies
101 Software Avenue,
Nanjing 210012
China
Phone: +86-25-56625375
Email: liyizhou@huawei.com

Muhammad Durrani
Brocade communications Systems, Inc
mdurrani@Brocade.com

Sujay Gupta
IP Infusion
RMZ Centennial
Mahadevapura Post
Bangalore - 560048
India
EMail: sujay.gupta@ipinfusion.com

Andrew Qu
MediaTec
Email: laodulaodu@gmail.com

Tao Han
Huawei Technologies
101 Software Avenue,
Nanjing 210012
China
Phone: +86-25-56623454
Email: billow.han@huawei.com

INTERNET-DRAFT Donald Eastlake
Updates: 7178 Huawei
Intended status: Proposed Standard Mohammed Umair
IPinfusion
Yizhou Li
Huawei
Expires: September 7, 2015 March 8, 2015

TRILL: RBridge Channel Tunnel Protocol
<draft-ietf-trill-channel-tunnel-04.txt>

Abstract

The IETF TRILL (Transparent Interconnection of Lots of Links) protocol includes an optional mechanism, called RBridge Channel and specified in RFC 7178, for the transmission of typed messages between TRILL switches in the same campus and between TRILL switches and end stations on the same link. This document specifies two optional extensions to the RBridge Channel protocol: (1) A standard method to tunnel a variety of payload types by encapsulating them in an RBridge Channel message; and (2) A method to support security facilities for RBridge Channel messages. This document updates RFC 7178.

Status of This Memo

Table of Contents

1. Introduction

The IETF TRILL base protocol [RFC6325] has been extended with an optional RBridge Channel [RFC7178] facility to support transmission of typed messages (for example BFD [RFC7175]) between two TRILL switches (RBridges) in the same campus and between RBridges and end stations on the same link. When sent between RBridges in the same campus, a TRILL Data packet with a TRILL header is used and the destination RBridge is indicated by nickname. When sent between a RBridge and an end station on the same link in either direction a native RBridge Channel messages [RFC7178] is used with no TRILL header and the destination port or ports are indicated by a MAC address. (There is no mechanism to stop end stations on the same link, from sending native RBridge Channel messages to each other; however, such use is outside the scope of this document.)

This document updates [RFC7178] and specifies extensions to RBridge Channel that provides two additional facilities as listed below. Implementation and use of each of these facilities is optional, except that there are two payload types that MUST be implemented. Both of these facilities can be used in the same packet.

(1) A standard method to tunnel a variety of payload types by encapsulating them in an RBridge Channel message.

(2) A method to provide security facilities for RBridge Channel messages.

In case of conflict between this document and [RFC7178], this document takes precedence.

1.1 Terminology and Acronyms

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].

This document uses terminology and acronyms defined in [RFC6325] and [RFC7178]. Some of these are repeated below for convenience along with additional terms and acronyms.

AES - Advanced Encryption Standard.

CCM - Counter with CBC-MAC

Data Label - VLAN or FGL.

DTLS - Datagram TLS [RFC6347].

FGL - Fine Grained Label [RFC7172].

HKDF - Hash based Key Derivation Function [RFC5869].

RBridge - An alternative term for a TRILL switch.

SHA - Secure Hash Algorithm [RFC6234].

TRILL - Transparent Interconnection of Lots of Links or Tunneled Routing in the Link Layer.

TRILL switch - A device that implements the TRILL protocol [RFC6325], sometimes referred to as an RBridge.

2. Channel Tunnel Packet Format

The general structure of an RBridge Channel message between two TRILL switches (RBridges) in the same campus is shown in Figure 1 below. The structure of a native RBridge Channel message sent between an RBridge and an end station on the same link, in either direction, is shown in Figure 2 and, compared with the first case, omits the TRILL Header, inner Ethernet addresses, and Data Label. A Protocol field in the RBridge Channel Header gives the type of RBridge Channel message and indicates how to interpret the Channel Protocol Specific Payload [RFC7178].

```
+-----------------------------------+
|            Link Header            |
+-----------------------------------+
|            TRILL Header           |
+-----------------------------------+
|      Inner Ethernet Addresses     |
+-----------------------------------+
|      Data Label (VLAN or FGL)     |
+-----------------------------------+
|       RBridge Channel Header      |
+-----------------------------------+
| Channel Protocol Specific Payload |
+-----------------------------------+
|   Link Trailer (FCS if Ethernet)  |
+-----------------------------------+
```

Figure 1. RBridge Channel Packet Structure

```
+-----------------------------------+
|        Ethernet Link Header       |
+-----------------------------------+
|       RBridge Channel Header      |
+-----------------------------------+
| Channel Protocol Specific Payload |
+-----------------------------------+
|                FCS                |
+-----------------------------------+
```

Figure 2. Native RBridge Channel Frame

The RBridge Channel Header looks like this:

```
     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |         0x8946                | CHV  |   Channel Protocol    |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |         Flags         |  ERR  |                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               /
    /                               Channel Protocol Specific Data  /
    /-+-+-+-+-+-                                                    /
```

Figure 3. RBridge Channel Header

where 0x8946 is the RBridge Channel Ethertype and CHV is the Channel Header Version, currently zero.

The extensions specified herein are in the form of an RBridge Channel protocol, the Channel Tunnel Protocol. Figure 4 below expands the RBridge Channel Header and Protocol Specific Payload above for the case of the Channel Tunnel Protocol.

```
     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
  RBridge Channel Header:
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |         0x8946                |  0x0  | Tunnel Protocol =tbd1 |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |         Flags         |  ERR  |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                                    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  Channel Tunnel Protocol Specific: | SubERR| RESV4 | SType | PType |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |  Security Information, variable length (0 length if SType = 0)
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
    |      Tunneled Data, variable length
    |  ...
```

Figure 4. Channel Tunnel Header Structure

The RBridge Channel Header field specific to the RBridge Channel Tunnel Protocol is the Protocol field. Its contents MUST be the value allocated for this purpose (see Section 6).

The RBridge Tunnel Channel Protocol Specific Data fields are as follows:

SubERR: This field provides further details when a Tunnel Channel error is indicated in the RBridge Channel ERR field. If ERR is zero, then SubERR MUST be sent as zero and ignored on receipt. See Section 5.

RESV4: This field MUST be sent as zero. If non-zero when received, this is an error condition (see Section 4).

SType: This field describes the type of security information and features, including keying material, being provided. See Section 4.

PType: Payload type. This describes the tunneled data. See Section 3 below.

Security Information: Variable length information. Length is zero if SType is zero. See Section 4.

The Channel Tunnel protocol is integrated with the RBridge Channel facility. Channel Tunnel errors are reported as if they were RBridge Channel errors, using newly allocated code points in the ERR field of the RBridge Channel Header supplemented by the SubERR field.

3. Tunnel Payload Types

The RBridge Channel Tunnel Protocol can carry a variety of payloads as indicated by the PType field. Values are shown in the table below with further explanation after the table.

PType	Section	Description
0		Reserved
1	3.1	Null
2	3.2	RBridge Channel message
3	3.3	TRILL Data packet
4	3.4	TRILL IS-IS packet
5	3.5	Ethernet Frame
6-14		(Available for assignment by IETF Review)
15		Reserved

Table 1. Payload Type Values

While implementation of the Channel Tunnel protocol is optional, if it is implemented PTypes 1 (Null) and 2 (RBridge Channel message) MUST be implemented. PTypes 3, 4, and 5 MAY be implemented. The processing of any particular Channel Protocol message and its payload depends on meeting local security and other policy at the destination TRILL switch or end station.

3.1 Null Payload

The Null payload type (PType=1) is intended to be used for testing or messages such as key negotiation or the like. It indicates that there is no payload. Any data after the Security Information fields is ignored. Any particular use of the Null Payload should specify what VLAN or priority should be used when relevant.

3.2 RBridge Channel Message Payload

A PType of 2 indicates that the payload of the Channel Tunnel message is an encapsulated RBridge Channel message without the initial RBridge Channel Ethertype. Typical reasons for sending an RBridge Channel message inside a Channel Tunnel message are to provide security services, such as authentication or encryption.

This payload type looks like the following:

```
       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
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |    RBridge-Channel (0x8946)   |  0x0  | Tunnel Protocol = tbd1|
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |          Flags        |  ERR  | SubERR| RESV4 | SType |  0x2  |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |  Possible Security information
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |  0x0  |  Channel Protocol     |          Flags        |  ERR  |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |         Channel Protocol Specific Data ...                    |
      |
```

Figure 5. Tunneled Channel Message Channel Tunnel Structure

3.3 TRILL Data Packet

A PType of 3 indicates that the payload of the Tunnel protocol message is an encapsulated TRILL Data packet as shown in the figure below. (There is no TRILL Ethertype before the inner TRILL Data packet because that is just part of the Ethernet link header for a TRILL Data packet, not part of the TRILL header itself. The Optional Flags Word is only present if the F bit in the TRILL Header is 1.) If this PType is implemented and the message meets local policy for acceptance, the tunneled TRILL Data packet is handled as if it had been received by the destination TRILL switch on the port where the Channel Tunnel message was received.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    RBridge-Channel (0x8946)   |  0x0  | Tunnel Protocol = tbd1|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Flags        |  ERR  | SubERR| RESV4 | SType |  0x3  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Possible Security information
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| V |A|C|M| RESV  |F| Hop Count |      Egress Nickname          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Ingress Nickname         |     Optional Flags Word       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Optional Flags Word (cont.)   |        Inner.MacDA            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Inner.MacDA continued                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Inner.MacSA                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Inner.MacSA (cont.)        |     Inner Data Label ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
| TRILL Data Packet payload
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
```

Figure 6. Nested TRILL Data Packet Channel Tunnel Structure

3.4 TRILL IS-IS Packet

A PType of 4 indicates that the payload of the Tunnel protocol message is an encapsulated TRILL IS-IS PDU packet without the initial L2-IS-IS Ethertype as shown in the figure below. If this PType is implemented, the tunneled TRILL IS-IS packet is processed by the destination RBridge if it meets local policy. One possible use is to expedite the receipt of a link state PDU by some TRILL switch or switches with an immediate requirement for the enclosed link state PDU. Any link local IS-IS PDU (Hello, CSNP, or PSNP [IS-IS]; MTU-probe, MTU-ack [RFC7176]; or circuit scoped FS-LSP, FS-CSNP or FS-PSNP [RFC7356]) received via this channel tunnel payload type MUST be discarded.

```
     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |    RBridge-Channel (0x8946)   |  0x0  | Tunnel Protocol = tbd1|
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |          Flags        |  ERR  | SubERR| RESV4 | SType |  0x4  |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |  Possible Security information
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
    |      0x83     | rest of IS-IS PDU
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
```

Figure 7. Tunneled TRILL IS-IS Packet Structure

3.5 Ethernet Frame

If PType is 5, the Tunnel Protocol payload is an Ethernet frame as might be received from or sent to an end station except that the tunneled Ethernet frame’s FCS is omitted, as shown in Figure 8. (There is still an overall FCS if the RBridge Channel message is being sent on an Ethernet link.) If this PType is implemented and the message meets local policy, the tunneled frame is handled as if it had been received on the port on which the Tunnel Protocol message was received.

The priority of the RBridge Channel message can be copied from the Ethernet frame VLAN tag, if one is present, except that priorities 6 or 7 SHOULD only be used for important control messages.

```
       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
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |    RBridge-Channel (0x8946)   |  0x0  | Tunnel Protocol = tbd1|
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |          Flags        |  ERR  | SubERR| RESV4 | SType |  0x5  |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |  Possible Security information
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |                             MacDA                             |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |         MacDA (cont.)         |             MacSA             |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |                          MacSA (cont.)                        |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      | Any Ethernet frame tagging...
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
      |  Ethernet frame payload...
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
```

Figure 8. Ethernet Frame Channel Tunnel Structure

In the case of a non-Ethernet link, such as a PPP link [RFC6361], the ports on the link are considered to have link local synthetic 48-bit MAC addresses constructed by concatenating three 16-bit quantities. This constructed address MAY be used as the MacSA and, if the RBridge Channel message is link local, the source TRILL switch will have the information to construct such a MAC address for the destination TRILL switch port and that MAC address MAY be used as the MacDA.

These MAC addresses are constructed as follows: 0xFEFF, the nickname of the TRILL switch used in TRILL Hellos sent on that port, and the Port ID that the TRILL switch has assigned to that port, as shown in Figure 9. (Both the nickname and Port ID of the port on which a TRILL Hello is sent appear in the Special VLANs and Flags sub-TLV [RFC7176] in that Hello.) The resulting MAC address has the Local bit on and the Group bit off [RFC7042]. Since end stations are connected to TRILL switches over Ethernet, there will be no end stations on a non-Ethernet link in a TRILL campus. Thus such synthetic MAC addresses cannot conflict on the link with a real Ethernet port address.

```
       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
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |            0xFEFF             |            Nickname           |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |            Port ID            |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 9. Synthetic MAC Address

4. Security, Keying, and Algorithms

The following table gives the assigned values of the SType field and their meaning.

SType	Section	Meaning
0	4.4	None
1	4.5	[RFC5310] Based Authentication
2	4.6	DTLS Based Security
3	4.7	[RFC5310] Based Encryption and Authentication
4-14		Available for assignment on IETF Review
15		Reserved

Table 3. SType Values

4.1 Basic Security Format

For all SType values except zero, the Security Information starts with a byte of flag bits and a byte of remaining length as follows:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
|A|E|    RESV   |     Size      |   More Info
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
```

Figure 12. Security Information Format

The fields are as follows:

A: Zero if authentication is not being provided. One if it is.

E: Zero if encryption is not being provided. One if it is.

RESV: Six reserved bits that MUST be sent as zero and ignored on receipt. In the future, meanings may be assigned to these bits and those meanings may differ for different STypes.

Size: The number of bytes, as an unsigned integer, of More Info in the Security Information after the Size byte itself.

More Info: Additional Security Information of length Size. Contents depends on the SType.

The A and E bits are intended as hints and to assist is debugging. They are not guaranteed to be correct. They can be interpreted as follows:

A E	Comments
0 0	Neither authentication nor encryption is being provided.
1 0	Authentication only. The payload should be parsable by a security ignorant receiver. The Size field permits skipping the More Info field.
0 1	Encryption only. Some form of opportunistic security [RFC7435].
1 1	Authentication and Encryption.

4.2 Authentication and Encryption Coverage

Authentication in the RBridge Channel case (see Figure 1) is computed across the inner Ethernet Addresses, Data Label, relevant Channel Tunnel header information, and the payload. To be more precise, the covered area starts with the byte immediately after the TRILL Header ingress nickname or optional flag word, if present, and extends to just before the TRILL Data packet link trailer, for example just before the FCS for Ethernet. If an authentication value is included in the Info field specified in Section 4.1, it is treated as zero when authentication is calculated. If an authentication value is included in a payload after the security information, it is calculated as provided by the SType and algorithms in use.

Authentication in the native RBridge Channel case (see Figure 2), is as specified in the above paragraph except that it starts with the RBridge Channel Ethertype, since there are no TRILL Header, inner Ethernet address, or Data Label.

If encryption is provided, it covers the payload from right after the Channel Tunnel header security information through to just before the TRILL Data packet link trailer.

4.3 Derived Keying Material

In some cases, it is possible to use keying material derived from [RFC5310] IS-IS keying material. In such cases, the More Info field shown in Section 4.1 includes a two byte Key ID to identify the IS-IS keying material. The keying material actually used in Channel Tunnel security is derived from the IS-IS keying material as follows:

```
HKDF-Expand-SHA256 ( IS-IS-key, "Channel Tunnel" | 0x0S, L )
```

where "|" indicates concatenation, HKDF is as in [RFC5869], SHA256 is as in [RFC6234],IS-IS-key is the input keying material, "Channel Tunnel" is the 14-character [RFC20] string indicated, 0x0S is a single byte where S is the SType for which this key derivation is being used, and L is the length of output keying material needed.

4.4 SType None

No security services are being invoked. The length of the Security Information field (see Figure 6) is zero.

4.5 RFC 5310 Based Authentication

The Security Information (see Figure 6) is the flags and Size bytes specified in Section 4.1 with the value of the [RFC5310] Key ID and Authentication Data as shown in Figure 13.

```
                     1 1 1 1 1 1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|1|0|    RESV   |     Size      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Key ID              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                               |
+
| Authentication Data (Variable)
+
|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
```

Figure 13. SType 1 Security Information

- o RESV: Six bits that MUST be sent as zero and ignored or receipt.
- o Size: Set to 2 + the size of Authentication Data in bytes.
- o Key ID: specifies the same keying value and authentication algorithm that that Key ID specifies for TRILL IS-IS LSP [RFC5310] Authentication TLVs. The keying material actually used is derived as shown in Section 4.3.
- o Authentication Data: The authentication data produced by the key and algorithm associated with the Key ID acting on the packet as specified in Section 4.2. Length of authentication data depends on the algorithm.

4.6 DTLS Based Security

DTLS supports key negotiation and provides both encryption and authentication. This optional SType in Channel Tunnel uses DTLS 1.2 [RFC6347]. It is intended for pairwise use. The presumption is that in the RBridge Channel case (Figure 1) the M bit in the TRILL Header would be zero and in the native RBridge Channel case (Figure 2), the Outer.MacDA would be individually addressed.

TRILL switches that implement the Channel Tunnel DTLS SType SHOULD support the use of certificates for DTLS. In this case the Size field shown in Section 4.1 MUST be zero and the Security Information is as shown in Figure 14.

Also, if they support certificates, they MUST support the following algorithm:

o TLS_RSA_WITH_AES_128_CBC_SHA256 [RFC5246]

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|1|1|   RESV    |       0       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 14. DTLS Cert or Special Pre-shared Key Security Information

TRILL switches that support the Channel Tunnel DTLS SType MUST support the use of pre-shared keys for DTLS. The Size field as shown in Section 4.1 MUST be either zero or 2. If Size is zero as shown in Figure 14, a pre-shared key specifically associated with Channel Tunnel DTLS is used. If Size is 2 as shown in Figure 15, a two byte [RFC5310] Key ID is present and the pre-shared key is derived from the secret key associated with that Key ID as shown in Section 4.3.

The following cryptographic algorithms MUST be supported for use with pre-shared keys:

o TLS_PSK_WITH_AES_128_CBC_SHA256 [RFC5487]

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|1|1|   RESV    |       2       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Key ID              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 15. DTLS Derived Pre-shared Key Security Information

When DTLS security is used, the entire payload of the Channel Tunnel packet, starting just after the Security Information and ending just before the link trailer, is a DTLS record [RFC6347].

4.7 RFC 5310 Based Encryption and Authentication

This SType is based on pre-existing [RFC5310] keying material but does not use any algorithm that may be associated with a Key ID under [RFC5310]. Instead it uses the derived key as specified in Section 4.3 with the algorithm specified by a Crypto Suite ID. Key negotiation is not provided and this SType is intended for multi-destination message use. The presumption is that in the RBridge Channel case (Figure 1) the M bit in the TRILL Header would be one and in the native RBridge Channel case (Figure 2), the Outer.MacDA would be group addressed.

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|1|1|   RESV    |       4       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Key ID              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Crypto Suite ID           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 16. DTLS Derived Pre-shared Key Security Information

4.7.1 Channel-Tunnel-CCM

The initially specified Crypto Suite has ID 0x0001, is called Channel-Tunnel-CCM (Channel Tunnel Counter with CBC-MAC), and is mandatory to implement if this SType is supported.

Channel-Tunnel-CCM is based on [RFC3610] using AES-128 as the encryption function. The minimum authentication field size permitted is 8 octets. There is additional authenticated data which is the authenticated data indicated in Section 4.2 up to but not including any of the Tunneled Data (Figure 4). The message size is limited to 2**16 - 2**8 bytes so the length of the length of message field is always 2 bytes. There are thus 13 bytes available for nonce [RFC3610]. Since it is possible that the same Key ID could be used by different TRILL switches, the nonce MUST include an identifier for the originating TRILL switch. It is RECOMMENDED that this be the first 6 bytes of its IS-IS System ID as these will be unique across the campus. The remaining 7 bytes (56 bits) need to be such that the nonce is always unique for a particular key, for example a counter for which care is taken that it is always incremented after each use and its value is preserved over TRILL switch crashes, re-starts, and the like. Should there be a danger of exhausting such a counter, the TRILL switch MUST take steps such as causing re-keying of the [RFC5310] key ID it is using and/or changing to use a different Key ID.

5. Channel Tunnel Errors

RBridge Channel Tunnel Protocol errors are reported like RBridge Channel level errors. The ERR field is set to one of the following error codes:

ERR	Meaning
6	Unknown or unsupported field value
7	Authentication failure
8	Error in nested RBridge Channel message
(more TBD?)	

Table 4. Additional ERR Values

5.1 SubERRs under ERR 6

If the ERR field is 6, the SubERR field indicates the problematic field or value as show in the table below.

SubERR	Meaning (for ERR = 6)
0	Non-zero RESV4 nibble
1	Unsupported SType
2	Unsupported PType
4	Unsupported crypto algorithm
5	Unknown Key ID
(more TBD)	

Table 5. SubERR values under ERR 6

5.2 Nested RBridge Channel Errors

If
  a Channel Tunnel message is sent with security and with a payload type (PType) indicating a nested RBridge Channel message
and
  there is an error in the processing of that nested message that results in a return RBridge Channel message with a non-zero ERR field,
then that returned message SHOULD also be nested in an Channel Tunnel message using the same type of security. In this case, the ERR field in the Channel Tunnel envelope is set to 8 indicating that there is a nested error being tunneled back.

6. IANA Considerations

IANA has assigned tbd1 as the RBridge Channel protocol number the "Channel Tunnel" protocol from the range assigned by Standards Action.

The added RBridge Channel protocols registry entry on the TRILL Parameters web page is as follows:

Protocol	Description	Reference
tbd1	Tunnel Channel	[this document]

7. Security Considerations

The RBridge Channel tunnel facility has potentially positive and negative effects on security.

On the positive side, it provides optional security that can be used to authenticate and/or encrypt RBridge Channel messages. Some RBridge Channel message payloads, such as BFD [RFC7175], provide their own security but where this is not true, consideration should be give to requiring use of the security features of the Tunnel Protocol.

On the negative side, the optional ability to tunnel various payload types and to tunnel them not just between TRILL switches but to and from end stations can increase risk unless precautions are taking. The processing of decapsulated Tunnel Protocol payloads is not a good place to be liberal in what you accept as the tunneling facility makes it easier for unexpected messages to pop up in unexpected places in a TRILL campus due to accidents or the actions of an adversary. Local policies should generally be strict and only process payload types required and then only with adequate authentication for the particular circumstances.

In connection with the use of DTLS for security as specified in Section 4.5, see [RFC7457].

See [RFC7178] for general RBridge Channel Security Considerations.

See [RFC6325] for general TRILL Security Considerations.

Normative References

[IS-IS] - ISO/IEC 10589:2002, Second Edition, "Information technology -- Telecommunications and information exchange between systems -- 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)", 2002.

[RFC20] - Cerf, V., "ASCII format for network interchange", STD 80, RFC 20, October 1969, <http://www.rfc-editor.org/info/rfc20>.

[RFC2119] - Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3610] - Whiting, D., Housley, R., and N. Ferguson, "Counter with CBC-MAC (CCM)", RFC 3610, September 2003, <http://www.rfc-editor.org/info/rfc3610>.

[RFC5246] - Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008, <http://www.rfc-editor.org/info/rfc5246>.

[RFC5310] - Bhatia, M., Manral, V., Li, T., Atkinson, R., White, R., and M. Fanto, "IS-IS Generic Cryptographic Authentication", RFC 5310, February 2009.

[RFC5487] - Badra, M., "Pre-Shared Key Cipher Suites for TLS with SHA-256/384 and AES Galois Counter Mode", RFC 5487, March 2009, <http://www.rfc-editor.org/info/rfc5487>.

[RFC5869] - Krawczyk, H. and P. Eronen, "HMAC-based Extract-and-Expand Key Derivation Function (HKDF)", RFC 5869, May 2010, <http://www.rfc-editor.org/info/rfc5869>.

[RFC6325] - Perlman, R., D. Eastlake, D. Dutt, S. Gai, and A. Ghanwani, "RBridges: Base Protocol Specification", RFC 6325, July 2011.

[RFC6347] - Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, January 2012, <http://www.rfc-editor.org/info/rfc6347>.

[RFC7172] - Eastlake 3rd, D., Zhang, M., Agarwal, P., Perlman, R., and D. Dutt, "Transparent Interconnection of Lots of Links (TRILL): Fine-Grained Labeling", RFC 7172, May 2014.

[RFC7176] - Eastlake 3rd, D., Senevirathne, T., Ghanwani, A., Dutt, D., and A. Banerjee, "Transparent Interconnection of Lots of Links (TRILL) Use of IS-IS", RFC 7176, May 2014,

<http://www.rfc-editor.org/info/rfc7176>.

[RFC7178] - Eastlake 3rd, D., Manral, V., Li, Y., Aldrin, S., and D. Ward, "Transparent Interconnection of Lots of Links (TRILL): RBridge Channel Support", RFC 7178, May 2014.

[RFC7356] - Ginsberg, L., Previdi, S., and Y. Yang, "IS-IS Flooding Scope Link State PDUs (LSPs)", RFC 7356, September 2014, <http://www.rfc-editor.org/info/rfc7356>.

[rfc7180bis] - Eastlake, D., Zhang, M., Perlman, R. Banerjee, A., Ghanwani, A., and S. Gupta, "TRILL: Clarifications, Corrections, and Updates", Draft-ietf-trill-rfc7180bis, work in progress.

Informative References

[RFC6234] - Eastlake 3rd, D. and T. Hansen, "US Secure Hash Algorithms (SHA and SHA-based HMAC and HKDF)", RFC 6234, May 2011.

[RFC6361] - Carlson, J. and D. Eastlake 3rd, "PPP Transparent Interconnection of Lots of Links (TRILL) Protocol Control Protocol", RFC 6361, August 2011

[RFC7042] - Eastlake 3rd, D. and J. Abley, "IANA Considerations and IETF Protocol and Documentation Usage for IEEE 802 Parameters", BCP 141, RFC 7042, October 2013.

[RFC7175] - Manral, V., Eastlake 3rd, D., Ward, D., and A. Banerjee, "Transparent Interconnection of Lots of Links (TRILL): Bidirectional Forwarding Detection (BFD) Support", RFC 7175, May 2014.

[RFC7435] - Dukhovni, V., "Opportunistic Security: Some Protection Most of the Time", RFC 7435, December 2014, <http://www.rfc-editor.org/info/rfc7435>.

[RFC7457] - Sheffer, Y., Holz, R., and P. Saint-Andre, "Summarizing Known Attacks on Transport Layer Security (TLS) and Datagram TLS (DTLS)", RFC 7457, February 2015, <http://www.rfc-editor.org/info/rfc7457>.

Appendix Z: Change History

From -00 to -01

1. Fix references for RFCs published, etc.

2. Explicitly mention in the Abstract and Introduction that this document updates [RFC7178].

3. Add this Change History Appendix.

From -01 to -02

1. Remove section on the "Scope" feature as mentioned in http://www.ietf.org/mail-archive/web/trill/current/msg06531.html

2. Editorial changes to IANA Considerations to correspond to draft-leiba-cotton-iana-5226bis-11.txt.

3. Improvements to the Ethernet frame payload type.

4. Other Editorial changes.

From -02 to -03

1. Update TRILL Header to correspond to [rfc7180bis].

2. Remove a few remnants of the "Scope" feature that was removed from -01 to -02.

3. Substantial changes to and expansion of Section 4 including adding details of DTLS security.

4. Updates and additions to the References.

5. Other minor editorial changes.

From -03 to -04

1. Add SType for [RFC5310] keying based security that provides encryption as well as authentication.

2. Editorial improvements and fixes.

Acknowledgements

The contributions of the following are hereby acknowledged:

TBD

The document was prepared in raw nroff. All macros used were defined within the source file.

Authors' Addresses

Donald E. Eastlake, 3rd
Huawei Technologies
155 Beaver Street
Milford, MA 01757 USA

Phone: +1-508-333-2270
EMail: d3e3e3@gmail.com

Mohammed Umair
IPinfusion

EMail: mohammed.umair2@gmail.com

Yizhou Li
Huawei Technologies
101 Software Avenue,
Nanjing 210012, China

Phone: +86-25-56622310
EMail: liyizhou@huawei.com

TRILL Working Group
Internet Draft
Intended status: Standard Track
Updates: 6325
Tissa Senevirathne
CISCO
Janardhanan Pathangi
DELL
Jon Hudson
Brocade

March 9, 2015
Expires: September 2015

Coordinated Multicast Trees (CMT) for TRILL
draft-ietf-trill-cmt-06.txt

Status of this Memo

Copyright Notice

Abstract

TRILL facilitates loop free connectivity to non-TRILL networks via choice of an Appointed Forwarder for a set of VLANs. Appointed Forwarders provide load sharing based on VLAN with an active-standby model. High performance applications require an active-active load sharing model as discussed in RFC 7379. The Active-Active load-sharing model can be accomplished by representing any given non-TRILL network with a single virtual RBridge. Virtual representation of the non-TRILL network with a single RBridge poses serious challenges in multi-destination RPF (Reverse Path Forwarding) check calculations. This document specifies required enhancements to build Coordinated Multicast Trees (CMT) within the TRILL campus to solve related RPF issues. CMT provides flexibility to RBridges in selecting desired path of association to a given TRILL multi-destination distribution tree. This document updates RFC 6325.

Table of Contents

1. Introduction

TRILL (Transparent Interconnection of Lots of Links) presented in [RFC6325] and other related documents, provides methods of utilizing all available paths for active forwarding, with minimum configuration. TRILL utilizes IS-IS (Intermediate System to Intermediate System [IS-IS]) as its control plane and uses a TRILL header with hop count.

[RFC6325], [RFC7177] and [RFC6439] provide methods for interoperability between TRILL and Ethernet end stations and bridged networks. [RFC6439], provide an active-standby solution, where only one of the RBridges on a link with end stations is in the active forwarding state for end station traffic for any given VLAN. That RBridge is referred to as the Appointed Forwarder (AF). All frames ingressed into a TRILL network via the Appointed Forwarder are encapsulated with the TRILL header with a nickname held by the ingress AF RBridge. Due to failures, re-configurations and other network dynamics, the Appointed Forwarder for any set of VLANs may change. RBridges maintain forwarding tables that contain destination MAC address and Data Label (VLAN or Fine Grained Label (FGL)) to egress RBridge binding. In the event of an AF change, forwarding tables of remote RBridges may continue to forward traffic to the previous AF and that traffic may get discarded at the egress, causing traffic disruption.

Mission critical applications such as High Performance Data Centers require resiliency during failover. The active-active forwarding model minimizes impact during failures and maximizes the available network bandwidth. A typical deployment scenario, depicted in Figure 1, may have either End Stations and/or Legacy bridges attached to the RBridges. These Legacy devices typically are multi-homed to several RBridges and treat all of the uplinks independently using a Local Active-Active Link Protocol (LAALP [RFC7379]) such as a single Multi-Chassis Link Aggregation (MC-LAG) bundle or Distributed Resilient Network Interconnect [8021AX]. The Appointed Forwarder designation presented in [RFC6439] requires each of the edge

RBridges to exchange TRILL Hello packets. By design, an LAALP does not forward packets received on one of the member ports of the MC-LAG to other member ports of the same MC-LAG. As a result the AF designation methods presented in [RFC6439] cannot be applied to deployment scenario depicted in Figure 1. [RFC7379]

An active-active load-sharing model can be implemented by representing the edge of the network connected to a specific edge group of RBridges by a single virtual RBridge. Each virtual RBridge MUST have a nickname unique within its TRILL campus. In addition to an active-active forwarding model, there may be other applications that may requires similar representations.

Sections 4.5.1 and 4.5.2 of [RFC6325] as updated by [RFC7180] specify distribution tree calculation and RPF (Reverse Path Forwarding) check calculation algorithms for multi-destination forwarding. These algorithms strictly depend on link cost and parent RBridge priority. As a result, based on the network topology, it may be possible that a given edge RBridge, if it is forwarding on behalf of the virtual RBridge, may not have a candidate multicast tree that the edge RBridge can forward traffic on because there is no tree for which the virtual RBridge is a leaf node from the edge RBridge.

In this document we present a method that allows RBridges to specify the path of association for real or virtual child nodes to distribution trees. Remote RBridges calculate their forwarding tables and derive the RPF for distribution trees based on the distribution tree association advertisements. In the absence of distribution tree association advertisements, remote RBridges derive the SPF (Shortest Path First) based on the algorithm specified in section 4.5.1 of [RFC6325] as updated by [RFC7180]. This document updates [RFC6325] by changing, when CMT sub-TLVs are present, [RFC6325]'s mandatory provisions as to how distribution tree are constructed.

Other applications, beside the above mentioned active-active forwarding model, may utilize the distribution tree association framework presented in this document to associate to distribution trees through a preferred path.

This proposal requires presence of multiple multi-destination trees within the TRILL campus and updating all the RBridges in the network to support the new Affinity sub-TLV (Section 3.). It is expected that both of these requirements will be met as they are control plane changes, and will be common deployment scenarios. In case either of the above two conditions are not met RBridges MUST support a fallback option for interoperability. Since the fallback is expected to be a temporary phenomenon till all RBridges are upgraded, this proposal gives guidelines for such fallbacks, and does not mandate or specify any specific set of fallback options.

1.1. Scope and Applicability

This document specifies an Affinity sub-TLV to solve RPF issues at the active-active edge. Specific methods in this document for making use of the Affinity sub-TLV are applicable where a virtual RBridge is used to represent multiple RBridges are connected to an edge CE through an LAALP such as multi-chassis link aggregation or some similar arrangement where the RBridges cannot see each other's Hellos.

This document DOES NOT provide other required operational elements to implement an active-active edge solution, such as methods of multi-chassis link aggregation. Solution specific operational elements are outside the scope of this document and will be covered in other documents. (See, for example [TRILLPN].)

Examples provided in this document are for illustration purposes only.

1.2. Contributors

The work in this document is a result of much passionate discussions and contributions from following individuals. Their names are listed in alphabetical order:

Ayan Banerjee, Dinesh Dutt, Donald Eastlake, Mingui Zhang, Radia Perlman, Sam Aldrin, Shivakumar Sundaram and Zhai Hongjun.

2. Conventions used in this document

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].

In this document, these words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying [RFC2119] significance.

2.1. Acronyms and Phrases

The following acronyms and phrases are used in this document:

AF: Appointed Forwarder [RFC6439].

CE : Customer Ethernet device, that is a device that performs forwarding based on 802.1Q bridging. This also can be end-station or a server.

Data Label: VLAN or FGL.

LAALP: Local Active-Active Link Protocol [RFC7379].

MC-LAG: . Multi-Chassis Link Aggregation is a proprietary extension to [8021AX], that facilitates connecting group of links from an originating device (A) to a group of discrete devices (B). Device (A) treats, all of the links in a given Multi-Chassis Link Aggregation bundle as a single logical interface and treats all devices in Group (B) as a single logical device for all forwarding purposes. Device (A) does not forward packets receive on Multi-Chassis Link bundle out of the same Multi-Chassis link bundle. Figure 1 depicts a specific use case example.

RPF: Reverse Path Forwarding. See section 4.5.2 of [RFC6325].

3. The AFFINITY sub-TLV

Association of an RBridge to a multi-destination distribution tree through a specific path is accomplished by using a new IS-IS sub-TLV, the Affinity sub-TLV.

The AFFINITY sub-TLV appears in Router Capability TLVs or MT Capability TLVs that are within LSP PDUs, as described in [RFC7176] which specifies the code point and data structure for the Affinity sub-TLV.

4. Multicast Tree Construction and Use of Affinity Sub-TLV

Figure 1 and Figure 2 below show the reference topology and a logical topology using CMT to provide active-active service.

```
              -------------------
             /                     \
            |                       |
            |   TRILL Campus        |
            |                       |
             \                     /
              --------------------
                 |       |    |
            -----        |      --------
           |             |              |
       +------+      +------+      +------+
       |      |      |      |      |      |
       |(RB1) |      |(RB2) |      | (RBk)|
       +------+      +------+      +------+
         |..|          |..|           |..|
         |  +----+     |  |           |  |
         |   +---|-----|--|-----------+  |
         | +-|---|-----+  +------------+ |
 MC-     | | |   +-------------------+ | |
 LAG--->(| | |)                     (| | |) <- MC-LAG
       +-------+    .  .  .        +-------+
       | CE1   |                   | CEn   |
       |       |                   |       |
       +-------+                   +-------+
```

Figure 1 Reference Topology

```
            ------------------              Sample Multicast Tree (T1)
           /                    \
          |                      |                  |
          | TRILL Campus         |                  o RBn
          |                      |                / | \
           \                    /                /  |  ---\
            ---------------------             RB1o  o      o
               |       |    |                    |   RB2    RBk
   |           |       |     ----------          |
               |       |              |         oRBv
           +------+ +------+      +------+
           |      | |      |      |   |
           |(RB1) | |(RB2) |      | (RBk)|
           +------+ +------+      +------+
               |..|       |..|             |..|
               |  +----+  |  |             |  |
               |   +---|--|--|-------------+  |
               | +-|---|--+  +--------------+ |
       MC-     | | |   +------------------+ | |
       LAG--->(| | |)                    (| | |) <- MC-LAG
              +-------+    .  .  .       +-------+
              | CE1   |                  | CEn   |
              |       |                  |       |
              +-------+                  +-------+
```

Figure 2 Example Logical Topology

4.1. Update to RFC 6325

Section 4.5.1 of [RFC6325], is updated to change the calculation of distribution trees as below:

Each RBridge that desires to be the parent RBridge for child Rbridge RBy in a multi-destination distribution tree x announces the desired association using an Affinity sub-TLV. The child RBridge RBy is specified by its nickname (or one of its nicknames if it holds more than one).

When such an Affinity sub-TLV is present, the association specified by the affinity sub-TLV MUST be used when constructing the multi-destination distribution tree except in case of conflicting Affinity sub-TLV which are resolved as specified in Section 5.3.   In the absence of such an Affinity sub-TLV, or if there are any RBridges in the campus that are do not support Affinity sub-TLV, distribution trees are calculated as specified in the section 4.5.1 of [RFC6325] as updated by [RFC7180]. Section 4.3. below specifies how to identify RBridges that support Affinity sub-TLV capability.

4.2. Announcing virtual RBridge nickname

Each edge RBridge RB1 to RBk advertises in its LSP virtual RBridge nickname RBv using the Nickname sub-TLV (6), [RFC7176], along with their regular nickname or nicknames.

It will be possible for any RBridge to determine that RBv is a virtual RBridge because each RBridge (RB1 to RBk) this appears to be advertising that it is holding RBv is also advertising an Affinity sub-TLV asking that RBv be its child in one or more trees.

Virtual RBridges are ignored when determining the distribution tree roots for the campus.

All RBridges outside the edge group assume that multi-destination packets with ingress nickname RBv might use any of the distribution trees that any member of the edge group is advertising that it might use.

4.3. Affinity Sub-TLV Capability.

RBridges that announce the TRILL version sub-TLV [RFC7176] and set the Affinity capability bit (Section 7.) support the Affinity sub-TLV and calculation of multi-destination distribution trees and RPF checks as specified herein.

5. Theory of operation

5.1. Distribution Tree provisioning

Let’s assume there are n distribution trees and k edge RBridges in the edge group of interest.

If n >= k

Let’s assume edge RBridges are sorted in numerically ascending order by IS-IS SystemID such that RB1 < RB2 < RBk. Each Rbridge in the numerically sorted list is assigned a monotonically increasing number j such that; RB1=0, RB2=1, RBi=j and RBi+1=j+1.

Assign each tree to RBi such that tree number { (tree_number) % k}+1 } is assigned to RBridge i for tree_number from 1 to n. where n is the number of trees, k is the number of RBridges considered for tree allocation, and ''%'' is the integer division remainder operation.

If n < k

Distribution trees are assigned to RBridges RB1 to RBn, using the same algorithm as n >= k case. RBridges RBn+1 to RBk do not participate in active-active forwarding process on behalf of RBv.

5.2. Affinity Sub-TLV advertisement

Each RBridge in the RB1 through RBk domain advertises an Affinity TLV for RBv to be its child.

As an example, let's assume that RB1 has chosen Trees t1 and tk+1 on behalf of RBv.

RB1 advertises affinity TLV; {RBv, Num of Trees=2, t1, tk+1.

Other RBridges in the RB1 through RBk edge group follow the same procedure.

5.3. Affinity sub-TLV conflict resolution

In TRILL, multi-destination distribution trees are built outward from the root. If an RBridge RB1 advertises an Affinity sub-TLV with an AFFINITY RECORD that asks for RBridge RBroot to be its child in a tree rooted at RBroot, that AFFINITY RECORD is in conflict with TRILL distribution tree root determination and MUST be ignored.

If an RBridge RB1 advertises an Affinity sub-TLV with an AFFINITY RECORD that's ask for nickname RBn to be its child in any tree and RB1 is not adjacent to a real or virtual RBridge RBn, that AFFINITY RECORD is in conflict with the campus topology and MUST be ignored.

If different RBridges advertise Affinity sub-TLVs that try to associate the same virtual RBridge as their child in the same tree or trees, those Affinity sub-TLVs are in conflict with each other for those trees. The nicknames of the conflicting RBridges are compared to identify which RBridge holds the nickname that is the highest priority to be a tree root, with the System ID as the tiebreaker

The RBridge with the highest priority to be a tree root will retain the Affinity association. Other RBridges with lower priority to be a tree root MUST stop advertising their conflicting Affinity sub-TLV, re-calculate the multicast tree affinity allocation, and, if appropriate, advertise a new non-conflicting Affinity sub-TLV.

Similarly, remote RBridges MUST honor the Affinity sub-TLV from the RBridge with the highest priority to be a tree root (use system-ID as the tie-breaker in the event of conflicting priorities) and ignore the conflicting Affinity sub-TLV entries advertised by the RBridges with lower priorities to be tree roots.

5.4. Ingress Multi-Destination Forwarding

If there is at least one tree on which RBv has affinity via RBk, then RBk performs the following operations, for multi-destination frames received from a CE node:

1. Flood to locally attached CE nodes subjected to VLAN and multicast pruning.
2. Ingress in the TRILL header and assign ingress RBridge nickname as RBv (nickname of the virtual RBridge).
3. Forward to one of the distribution trees, tree x in which RBv is associated with RBk.

5.4.1. Forwarding when n < k

If there is no tree on which RBv can claim affinity via RBk (probably because the number of trees n built is less than number of RBridges k announcing the affinity sub-TLV), then RBk MUST fall back to one of the following

1. This RBridge should stop forwarding frames from the CE nodes, and should mark that port as disabled. This will prevent CE nodes from forwarding data on to this RBridge, and only use those RBridges which have been assigned a tree -

   -OR-
2. This RBridge tunnels multi-destination frames received from attached native devices to an RBridge RBy that has an assigned tree. The tunnel destination should forward it to the TRILL network, and also to its local access links. (The mechanism of tunneling and handshake between the tunnel source and

destination are out of scope of this specification and may be addressed in other documents such as [ChannelTunnel].)

Above fallback options may be specific to active-active forwarding scenario. However, as stated above, Affinity sub-TLV may be used in other applications. In such event the application SHOULD specify applicable fallback options.

5.5. Egress Multi-Destination Forwarding

5.5.1. Traffic Arriving on an assigned Tree to RBk-RBv

Multi-destination frames arriving at RBk on a Tree x, where RBk has announced the affinity of RBv via x, MUST be forwarded to CE members of RBv that are in the frame’s VLAN. Forwarding to other end-nodes and RBridges that are not part of the network represented by the RBv virtual RBridge MUST follow the forwarding rules specified in [RFC6325].

5.5.2. Traffic Arriving on other Trees

Multi-destination frames arriving at RBk on a Tree y, where RBk has not announced the affinity of RBv via y, MUST NOT be forwarded to CE members of RBv. Forwarding to other end-nodes and RBridges that are not part of the network represented by the RBv virtual RBridge MUST follow the forwarding rules specified in [RFC6325].

5.6. Failure scenarios

The below failure recovery algorithm is presented only as a guideline. Implementations MAY include other failure recover algorithms. Details of such algorithms are outside the scope of this document.

5.6.1. Edge RBridge RBk failure

Each of the member RBridges of given virtual RBridge edge group is aware of its member RBridges through configuration, LSP advertisements, or some other method.

Member RBridges detect nodal failure of a member RBridge through IS-IS LSP advertisements or lack thereof.

Upon detecting a member failure, each of the member RBridges of the RBv edge group start recovery timer T_rec for failed RBridge RBi. If the previously failed RBridge RBi has not recovered after the expiry of timer T_rec, members RBridges perform the distribution tree assignment algorithm specified in section 5.1. Each of the member RBridges re-advertises the Affinity sub-TLV with new tree assignment. This action causes the campus to update the tree calculation with the new assignment.

RBi upon start-up, starts advertising its presence through IS-IS LSPs and starts a timer T_i. Member RBridges detecting the presence of RBi start a timer T_j. Timer T_j SHOULD be at least < T_i/2. (Please see note below)

Upon expiry of timer T_j, member RBridges recalculate the multi-destination tree assignment and advertised the related trees using Affinity sub-TLV.

Upon expiry of timer T_i, RBi recalculate the multi-destination tree assignment and advertises the related trees using Affinity TLV.

Note: Timers T_i and T_j are designed so as to minimize traffic down time and avoid multi-destination packet duplication.

5.7. Backward compatibility

Implementations MUST support backward compatibility mode to interoperate with pre Affinity sub-TLV RBRidges in the network. Such backward compatibility operation MAY include, however is not limited to, tunneling and/or active-standby modes of operations.

Example:

Step 1. Stop using virtual RBridge nickname for traffic ingressing from CE nodes

Step 2. Stop performing active-active forwarding. And fall back to active standby forwarding, based on locally defined policies. Definition of such policies is outside the scope of this document and may be addressed in other documents.

6. Security Considerations

In general, the RBridges in a campus are trusted routers and the authenticity of their link state information (LSPs) and link local PDUs (Hellos, etc.) can be enforced using regular IS-IS security mechanisms [IS-IS] [RFC5310]. This including authenticating the contents of the PDUs used to transport Affinity sub-TLVs.

The particular Security Considerations involve with different applications of the Affinity sub-TLV will be covered in the document(s) specifying those applications.

For general TRILL Security Considerations, see [RFC6325].

7. IANA Considerations

This document requires no IANA actions because the ''Affinity Supported'' capability bit and the Affinity sub-TLV have been assigned in [RFC7176].

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC5310] Bhatia, M., et.al. ''IS-IS Generic Cryptographic Authentication'', RFC 5310, February 2009.

[RFC6325] Perlman, R., et.al. ''RBridge: Base Protocol Specification'', RFC 6325, July 2011.

[RFC7177] Eastlake 3rf, D. et.al., ''RBridge: Adjacency'', RFC 7177, May 2014.

[RFC6439] Eastlake 3rd, D. et.al., ''RBridge: Appointed Forwarder'', RFC 6439, November 2011.

[RFC7176] Eastlake 3rd, D. et.al., ''Transparent Interconnection of Lots of Links (TRILL) Use of IS-IS'', RFC 7176, May 2014.

[RFC7180] Eastlake 3rd, D. et.al., ''TRILL: Clarifications, Corrections, and Updates'', RFC 7180, May 2014.

[IS-IS] ISO/IEC, ''Intermediate System to Intermediate System Routing Information Exchange Protocol for use in Conjunction with the Protocol for Providing the Connectionless-mode Network Service (ISO 8473)'' ISO/IEC 10589:2002.

8.2. Informative References

[RFC7379] Li, Y., Hao, W., Perlman, R., Hudson, J., and H. Zhai, "Problem Statement and Goals for Active-Active Connection at the Transparent Interconnection of Lots of Links (TRILL) Edge", RFC 7379, October 2014, <http://www.rfc-editor.org/info/rfc7379>.

[TRILLPN] Zhai,H., et.al ''RBridge: Pseudonode Nickname'', draft-hu-trill-pseudonode-nickname, Work in progress, November 2011.

[8021AX] IEEE, ''Link Aggregration'', IEEE Std 802.1AX-2014, December 2014.

[ChannelTunnel] D. Eastlake and Y. Li, "TRILL: RBridge Channel Tunnel Protocol", draft-ietf-trill-channel-tunnel, work in progress.

9. Acknowledgments

Authors wish to extend their appreciations towards individuals who volunteered to review and comment on the work presented in this document and provided constructive and critical feedback. Specific acknowledgements are due for Anoop Ghanwani, Ronak Desai, and Varun Shah. Very special Thanks to Donald Eastlake for his careful review and constructive comments.

This document was prepared using 2-Word-v2.0.template.dot.

Appendix A. Change History.

From -01 to -02:

Replaced all references to ''LAG'' with references to Multi-Chassis (MC-LAG) or the like.

Expanded, Security Considerations section.

Other editorial changes.

From -02 to -03

Minor editorial changes

From -03 to -04

Minor editorial changes and version update.

From -04 to -05

Editorial, reference, and other minor changes based on Document Shepherd review.

Authors' Addresses

Tissa Senevirathne
Cisco Systems
375 East Tasman Drive,
San Jose, CA 95134

Phone: +1-408-853-2291
Email: tsenevir@cisco.com

Janardhanan Pathangi
Dell/Force10 Networks
Olympia Technology Park,
Guindy Chennai 600 032

Phone: +91 44 4220 8400
Email: Pathangi_Janardhanan@Dell.com

Jon Hudson
Brocade
130 Holger Way
San Jose, CA 95134 USA

Email: jon.hudson@gmail.com

INTERNET-DRAFT
Intended status: Proposed Standard
Linda Dunbar
Donald Eastlake
Huawei
Radia Perlman
EMC
Igor Gashinsky
Yahoo
Yizhou Li
Huawei
Expires: May 9, 2014
November 10, 2014

TRILL: Edge Directory Assist Mechanisms
<draft-ietf-trill-directory-assist-mechanisms-01.txt>

Abstract

This document describes mechanisms for providing directory service to TRILL (Transparent Interconnection of Lots of Links) edge switches. The directory information provided can be used in reducing multi-destination traffic, particularly ARP/ND and unknown unicast flooding.

Status of This Memo

Table of Contents

1. Introduction

[RFC7067] gives a problem statement and high level design for using directory servers to assist TRILL [RFC6325] edge nodes in reducing multi-destination ARP/ND, reducing unknown unicast flooding traffic, and improving security against address spoofing within a TRILL campus. Because multi-destination traffic becomes an increasing burden as a network scales up in number of nodes, reducing ARP/ND and unknown unicast flooding improves TRILL network scalability. This document describes specific mechanisms for directory servers to assist TRILL edge nodes. These mechanisms are optional to implement.

The information held by the Directory(s) is address mapping and reachability information. Most commonly, what MAC address [RFC7042] corresponds to an IP address within a Data Label (VLAN or FGL (Fine Grained Label [RFC7172])) and the egress TRILL switch (RBridge), and optionally what specific TRILL switch port, from which that MAC address is reachable. But it could be what IP address corresponds to a MAC address or possibly other address mappings or reachability.

In the data center environment, it is common for orchestration software to know and control where all the IP addresses, MAC addresses, and VLANs/tenants are in a data center. Thus such orchestration software can be appropriate for providing the directory function or for supplying the Directory(s) with directory information.

Directory services can be offered in a Push or Pull Mode [RFC7067]. Push Mode, in which a directory server pushes information to TRILL switches indicating interest, is specified in Section 2. Pull Mode, in which a TRILL switch queries a server for the information it wants, is specified in Section 3. More detail on modes of operation, including hybrid Push/Pull, are provided in Section 4.

The mechanism used to initially populate directory data in primary servers is beyond the scope of this document. A primary server can use the Push Directory service to provide directory data to secondary servers as described in Section 2.5.

1.1 Uses of Directory Information

A TRILL switch can consult Directory information whenever it wants, by (1) searching through information that has been retained after being pushed to it or pulled by it or (2) by requesting information from a Pull Directory. However, the following are expected to be the most common circumstances leading to directory information use. All of these are cases of ingressing (or originating) a native frame.

1. ARP requests and replies [RFC826] are normally broadcast. But a directory assisted edge TRILL switches could intercept ARP messages and reply if the TRILL switch has the relevant information.

2. IPv6 ND (Neighbor Discovery [RFC4861]) requests and replies are normally multicast. Except in the case of Secure ND [RFC3971] where possession of the right keying material might be required, directory assisted edge TRILL switches could intercept ND messages and reply if the TRILL switch has the relevant information.

3. Unknown destination MAC addresses. An edge TRILL switch ingressing a native frame necessarily has to determine if it knows the egress RBridge from which the destination MAC address of the frame (in the frame’s VLAN or Fine Grained Label) is reachable. It might learn that information from the directory or could query the directory if it does not know. Furthermore, if the edge TRILL switch has complete directory information, it can detect forged source MAC address on the native frame and discard the frame in that case.

4. RARP [RFC903] is similar to ARP as above.

1.2 Terminology

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 RFC 2119 [RFC2119].

The terminology and acronyms of [RFC6325] are used herein along with the following:

COP: Complete Push flag bit. See Sections 2 and 6.1 below.

CSNP Time: Complete Sequence Number PDU Time. See ESDADI [RFC7357] and Section 6.1 below.

Data Label: VLAN or FGL.

FGL: Fine Grained Label [RFC7172].

Host: Application running on a physical server or a virtual machine. A host must have a MAC address and usually has at least one IP address.

IP: Internet Protocol. In this document, IP includes both IPv4 and IPv6.

PSH: Push Directory flag bit. See Sections 2 and 6.1 below.

PUL: Pull Directory flag bit. See Sections 3 and 6.3 below.

primary server: A Directory server that obtains the information it is serving up by a reliable mechanism outside the scope of this document designed to assure the freshness of that information. (See secondary server.)

RBridge: An alternative name for a TRILL switch.

secondary server: A Directory server that obtains the information it is serving up from one or more primary servers.

tenant: Sometimes used as a synonym for FGL.

TRILL switch: A device that implements the TRILL protocol.

2. Push Model Directory Assistance Mechanisms

In the Push Model [RFC7067], one or more Push Directory servers reside at TRILL switches and push down the address mapping information for the various addresses associated with end station interfaces and the TRILL switches from which those interfaces are reachable [IA]. This service is scoped by Data Label (VLAN or FGL [RFC7172]). A Push Directory also advertises whether or not it believes it has pushed complete mapping information for a Data Label. It might be pushing only a subset of the mapping and/or reachability information for a Data Label. The Push Model uses the ESADI [RFC7357] protocol as its distribution mechanism.

With the Push Model, if complete address mapping information for a Data Label being pushed is available, a TRILL switch (RBridge) which has that complete pushed information and is ingressing a native frame can simply drop the frame if the destination unicast MAC address can’t be found in the mapping information available, instead of flooding the frame (ingressing it as an unknown MAC destination TRILL Data frame). But this will result in lost traffic if ingress TRILL switch’s directory information is incomplete.

2.1 Requesting Push Service

In the Push Model, it is necessary to have a way for a TRILL switch to request information from the directory server(s). TRILL switches simply use the ESADI [RFC7357] protocol mechanism to announce, in their core IS-IS LSPs, the Data Labels for which they are participating in ESADI by using the Interested VLANs and/or Interested Labels sub-TLVs [RFC7176]. This will cause them to be pushed the Directory information for all such Data Labels that are being served by one or more Push Directory servers.

2.2 Push Directory Servers

Push Directory servers advertise their availability to push the mapping information for a particular Data Label to each other and to ESADI participants for that Data Label through ESADI by turning on the a flag bit in their ESADI Parameter APPsub-TLV for that ESADI instance (see [RFC7357] and Section 6.1). Each Push Directory server MUST participate in ESADI for the Data Labels for which it will push mappings and set the PSH (Push Directory) bit in its ESADI-Parameters APPsub-TLV for that Data Label.

For robustness, it is useful to have more than one copy of the data being pushed. Each Push Directory server is configured with a number

N in the range 1 to 8, which defaults to 2, for each Data Label for which it can push directory information. If the Push Directories for a Data Label are configured the same in this regard and enough such servers are available, N copies of the directory that will be pushed.

Each Push Directory server also has an 8-bit priority to be Active (see Section 6.1 of this document). This priority is treated as an unsigned integer where larger magnitude means higher priority and is in its ESADI Parameter APPsub-TLV. In cases of equal priority, the 6-byte IS-IS System IDs of the tied Push Directories are used as a tie breaker and treated as an unsigned integer where larger magnitude means higher priority.

For each Data Label it can serve, each Push Directory server orders, by priority, the Push Directory servers that it can see in the ESADI link state database for that Data Label that are data reachable [RFC7180] and determines its own position in that order. If a Push Directory server is configured to believe that N copies of the mappings for a Data Label should be pushed and finds that it is number K in the priority ordering (where number 1 is highest priority and number K is lowest), then if K is less than or equal to N the Push Directory server is Active. If K is greater than N it is Passive. Active and Passive behavior are specified below.

For a Push Directory to reside on an end station, one or more TRILL switches locally connected to that end station must proxy for the Push Directory server and advertise themselves as Push Directory servers. It appears to the rest of the TRILL campus that these TRILL switches (that are proxying for the end station) are the Push Directory server(s). The protocol between such a Push Directory end station and the one or more proxying TRILL switches acting as Push Directory servers is beyond the scope of this document.

2.3 Push Directory Server State Machine

The subsections below describe the states, events, and corresponding actions for Push Directory servers.

2.3.1 Push Directory States

A Push Directory Server is in one of six states, as listed below, for each Data Label it can serve. In addition, it has an internal State-Transition-Time variable for each Data Label it can serve which is set at each state transition and which enables it to determine how long it has been in its current state for that Data Label.

Down: A completely shut down virtual state defined for convenience in specifying state diagrams. A Push Directory Server in this state does not advertise any Push Directory data. It may be participating in ESDADI [RFC7357] with the PSH bit zero in its ESADI-Parameters or might be not participating in ESADI at all. All states other than the Down state are considered to be Up states.

Passive: No Push Directory data is advertised. Any outstanding EASDI-LSP fragments containing directory data are updated to remove that data and if the result is an empty fragment (contains nothing except possibly an Authentication TLV), the fragment is purged. The Push Directory participates in ESDADI [RFC7357] and advertises its ESADI fragment zero that includes an ESADI-Parameters APPsub-TLV with the PSH bit set to one and COP (Complete Push) bit zero.

Active: If a Push Directory server is Active, it advertises its directory data and any changes through ESADI [RFC7357] in its ESADI-LSPs using the Interface Addresses [IA] APPsub-TLV and updates that information as it changes. The PSH bit is set to one in the ESADI-Parameters and the COP bit set to zero.

Completing: Same behavior as the Active state but responds differently to events.

Complete: The same behavior as Active except that the COP bit in the ESADI-Parameters APPsub-TLV is set to one and the server responds differently to events.

Reducing: The same behavior as Complete but responds differently to events. The PSH bit remains a one but the COP bit is cleared to zero in the ESADI-Parameters APPsub-TLV. Directory updates continue to be advertised.

2.3.2 Push Directory Events and Conditions

Three auxiliary conditions referenced later in this section are defined as follows for convenience:

The Activate Condition: The Push Directory server determines that it is priority K among the data reachable Push Directory servers (where highest priority is 1), the server is configured that there should be N copies pushed, and K is less than or equal to N. For example, the Push Directory server is configured that 2 copies should be pushed and finds that it is priority 1 or 2 among the Push Directory servers it can see.

The Pacify Condition: The Push Directory server determines that it is priority K among the data reachable data reachable Push Directory servers (where highest priority is 1), the server is configured that there should be N copies pushed, and K is greater than N. For example, the Push Directory server is configured that 2 copies should be pushed and finds that it is priority 3 or lower priority (higher number) among the Push directory servers it can see.

The Time Condition: The Push Directory server has been in its current state for an amount of time equal to or larger than its CSNP time (see Section 6.1).)

The events and conditions listed below cause state transitions in Push Directory servers.

1. Push Directory server was Down but is now up.

2. The Push Directory server or the TRILL switch on which it resides is being shut down.

3. The Activate Condition is met and the server is not configured to believe it has complete data.

4. The Pacify Condition is met.

5. The Activate Condition is met and the server is configured to believe it has complete data.

6. The server is configured to believe it does not have complete data.

7. The Time Condition is met.

2.3.3 State Transition Diagram and Table

The state transition table is as follows:

Event	Down	Passive	Active	Completing	Complete	Reducing
1	Passive	Passive	Active	Completing	Complete	Reducing
2	Down	Down	Passive	Passive	Reducing	Reducing
3	Down	Active	Active	Active	Reducing	Reducing
4	Down	Passive	Passive	Passive	Reducing	Reducing
5	Down	Completing	Complete	Completing	Complete	Complete
6	Down	Passive	Active	Active	Reducing	Reducing
7	Down	Passive	Active	Complete	Complete	Active

The above state table is equivalent to the following transition diagram:

```
+-----------+
| Down      |<---------+
+-----------+          |
  |1  ^   | 3,4,5,6,7  |
  |   |   +------------+
  V   |2
+-----------+
| Passive   |<-----------------------
+-----------+        ^   ^         ^
  |5   |3  |1,4,6,7  |   |         |
  |    |   +---------+   |         |
  |    V                 |2,4       |
  |  +---------------------+         |
  |  | Active              |<--+   |
  |  +---------------------+   |   |
  |   |5  ^    |1,3,6,7  ^     |   |
  |   |   |    |         |     |   |
  |   |   |    +---------+     |   |
  |   |   |                    |   |
  V   V   |3,6                 |   |
+--------------+               |   |
| Completing   |-------------------+
+--------------+ 2,4           |
  |7  |1,5  ^                  |
  |   |     |                  |
  |   +-----+                  |
  V                            |7
+-------------+          +----------------+
| Complete    |--------->| Reducing       |<--+
+-------------+ 2,3,4,6  +----------------+   |
  |1,5,7 ^  ^              |5  |1,2,3,4,6      |
  |      |  |              |   |              |
  +------+  +--------------+   +--------------+
```

Figure 1. Push Server State Diagram

2.4 Additional Push Details

Push Directory mappings can be distinguished for other data distributed through ESADI because mappings are distributed only with the Interface Addresses APPsub-TLV [IA] and are flagged as being Push Directory data.

TRILL switches, whether or not they are a Push Directory server, MAY continue to advertise any locally learned MAC attachment information in ESDADI [RFC7357] using the Reachable MAC Addresses TLV [RFC6165]. However, if a Data Label is being served by complete Push Directory servers, advertising such locally learned MAC attachment generally

SHOULD NOT be done as it would not add anything and would just waste bandwidth and ESADI link state space. An exception might be when a TRILL switch learns local MAC connectivity and that information appears to be missing from the directory mapping.

Because a Push Directory server needs to advertise interest in one or more Data Labels even if it does not want to receive end station multidestination data in those Data Labels, the No Data (NOD) flag bit is provided as specified in Section 6.3.

When a Push Directory server is no longer data reachable [RFC7180], TRILL switches MUST ignore any Push Directory data from that server because it is no longer being updated and may be stale.

The nature of dynamic distributed asynchronous systems is such that it is impossible for a TRILL switch receiving Push Directory information to be absolutely certain that it has complete information. However, it can obtain a reasonable assurance of complete information by requiring two conditions to be met:

1. The PSH and COP bits are on in the ESADI zero fragment from the server for the relevant Data Label.
2. It has had continuous data connectivity to the server for the larger of the client’s and the server’s CSNP times.

Condition 2 is necessary because a client TRILL switch might be just coming up and receive an EASDI LSP meeting the requirement in condition 1 above but have not yet received all of the ESADI LSP fragment from the Push Directory server.

There may be conflicts between mapping information from different Push Directory servers or conflicts between locally learned information and information received from a Push Directory server. In case of such conflicts, information with a higher confidence value [RFC6325] is preferred over information with a lower confidence. In case of equal confidence, Push Directory information is preferred to locally learned information and if information from Push Directory servers conflicts, the information from the higher priority Push Directory server is preferred.

2.5 Primary to Secondary Server Push Service

A secondary Push or Pull Directory server is one that obtains its data from a primary directory server. Other techniques MAY be used but, by default, this data transfer occurs through the primary server acting as a Push Directory server for the Data Labels involved while the secondary directory server takes the pushed data it receives from the highest priority Push Directory server and re-originates it. Such a secondary server may be a Push Directory server or a Pull Directory server or both for any particular Data Label.

3. Pull Model Directory Assistance Mechanisms

In the Pull Model [RFC7067], a TRILL switch (RBridge) pulls directory information from an appropriate Directory Server when needed.

Pull Directory servers for a particular Data Label X are found by looking in the core TRILL IS-IS link state database for data reachable TRILL switches that advertise themselves by having the Pull Directory flag (PUL) on in their Interested VLANs or Interested Labels sub-TLV [RFC7176] for that Data Label. If multiple such TRILL switches indicate that they are Pull Directory Servers for a particular Data Label, pull requests can be sent to any one or more of them but it is RECOMMENDED that pull requests be preferentially sent to the server or servers that are lower cost from the requesting TRILL switch.

Pull Directory requests are sent by enclosing them in an RBridge Channel [RFC7178] message using the Pull Directory channel protocol number (see Section 6.2). Responses are returned in an RBridge Channel message using the same channel protocol number. See Section 3.2 for Query and Response message formats. For cache consistency or notification purposes, Pull Directory servers can sent unsolicited Update messages to client TRILL switches they believe may be holding old data and those clients can acknowledge such updates, as described in Section 3.3. All these messages have a common header as described in Section 3.1. Errors returns can be sent for queries or updates as described in Section 3.5.

The requests to Pull Directory Servers are typically derived from ingressed ARP [RFC826], ND [RFC4861], or RARP [RFC903] messages, or data frames with unknown unicast destination MAC addresses, intercepted by an ingress TRILL switch as described in Section 4.

Pull Directory responses include an amount of time for which the response should be considered valid. This includes negative responses that indicate no data is available. Thus both positive responses with data and negative responses can be cached and used to locally handle ARP, ND, RARP, unknown destination MAC frames, or the like, until the responses expire. If information previously pulled is about to expire, a TRILL switch MAY try to refresh it by issuing a new pull request but, to avoid unnecessary requests, SHOULD NOT do so if it has not been recently used. The validity timer of cached Pull Directory responses is NOT reset or extended merely because that cache entry is used.

3.1 Pull Directory Message Common Format

All Pull Directory messages are transmitted as the payload of RBridge Channel messages. All Pull Directory messages are formatted as described below starting with the following common 8-byte header:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Ver  | Type  | Flags | Count |      Err      |    SubErr     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Sequence Number                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Type Specific Payload - variable length
+-+-+-+- ...
```

Ver: Version of the Pull Directory protocol as an unsigned integer. Version zero is specified in this document.

Type: The Pull Directory message type as follows:

Type	Section	Name
0	3.2.1	Query
1	3.2.2	Response
2	3.1.4	Update
3	3.1.5	Acknowledge
4-15	-	Reserved

Flags: Four flag bits whose meaning depends on the Pull Directory message Type. Flags whose meaning is not specified are reserved, MUST be sent as zero, and MUST be ignored on receipt.

Count: Most Pull Directory message types specified herein have zero or more occurrences of a Record as part of the type specific payload. The Count field is the number of occurrences of that Record as an unsigned integer. For Pull Directory messages not structured with such occurrences, this field MUST be sent as zero and ignored on receipt.

Err, SubErr: The error and suberror fields are only used in messages that are in the nature of replies or acknowledgements. In messages that are requests or updates, these fields MUST be sent as zero and ignored on receipt. The meaning of values in the Err field depends on the Pull Directory message Type but in all cases the value zero means no error. The meaning of values in the SubErr field depends on both the message Type and on the value of the Err field but in all cases, a zero SubErr field is allowed and provides no additional information beyond the value of the Err field.

Sequence Number: An opaque 32-bit quantity set by the TRILL switch sending a request or other unsolicited message and returned in every corresponding reply or acknowledgement. It is used to match up responses with the message to which they respond.

Type Specific Payload: Format depends on the Pull Directory message Type.

3.2 Pull Directory Query and Response Messages

3.2.1 Pull Directory Query Message Format

A Pull Directory Query message is sent as the Channel Protocol specific content of an RBridge Channel message [RFC7178] TRILL Data packet or as a native RBridge Channel data frame (see Section 3.4). The Data Label of the packet is the Data Label in which the query is being made. The priority of the channel message is a mapping of the priority of the frame being ingressed that caused the query with the default mapping depending, per Data Label, on the strategy (see Section 4) or a configured priority for generated queries. (Geerate queries are those not the result of a mapping. For example, a query to refresh a cache entry.) The Channel Protocol specific data is formatted as a header and a sequence of zero or more QUERY Records as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Ver  | Type  | Flags | Count |      Err      |    SubErr     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Sequence Number                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| QUERY 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
| QUERY 2
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
| ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
| QUERY K
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
```

Ver, Sequence Number: See 3.1.

Type: 1 for Query. Queries received by an TRILL switch that is not a Pull Directory result in an error response (see Section 3.5) unless inhibited by rate limiting.

Flags, Err, and SubErr: MUST be sent as zero and ignored on receipt.

Count: Number of QUERY Records present. A Query message Count of zero is explicitly allowed, for the purpose of pinging a Pull Directory server to see if it is responding. On receipt of such an empty Query message, a Response message that also has a Count of zero is sent unless inhibited by rate limiting.

QUERY: Each QUERY Record within a Pull Directory Query message is formatted as follows:

```
    0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
  +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
  |        SIZE           |    RESV   |  QTYPE    |
  +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
If QTYPE = 1
  +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
  |                      AFN                      |
  +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
  |  Query address ...
  +--+--+--+--+--+--+--+--+--+--+--+--+...
If QTYPE = 2, 3, 4, or 5
  +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
  |  Query frame ...
  +--+--+--+--+--+--+--+--+--+--+--+--+...
```

SIZE: Size of the QUERY record in bytes as an unsigned integer starting not counting the SIZE field and following byte. Thus the minimum legal value is 2. A value of SIZE less than 2 indicates a malformed QUERY record. The QUERY record with the illegal SIZE value and any subsequent QUERY records MUST be ignored and the entire Query message MAY be ignored.

RESV: A block of reserved bits. MUST be sent as zero and ignored on receipt.

QTYPE: There are several types of QUERY Records currently defined in two classes as follows: (1) a QUERY Record that provides an explicit address and asks for all addresses for the interface specified by the query address and (2) a QUERY Record that includes a frame. The fields of each are specified below. Values of QTYPE are as follows:

QTYPE	Description
0	reserved
1	address query
2	ARP query frame
3	ND query frame
4	RARP query frame
5	Unknown unicast MAC query frame
6-14	assignable by IETF Review
15	reserved

AFN: Address Family Number of the query address.

Address Query: The query is asking for any other addresses, and the nickname of the TRILL switch from which they are reachable, that correspond to the same interface, within the data label of the query. Typically that would be either (1) a MAC address with the querying TRILL switch primarily interested in the TRILL switch by which that MAC address is reachable, or (2) an IP address with the querying TRILL switch interested in the corresponding MAC address and the TRILL switch by which that MAC address is reachable. But it could be some other address type.

Query Frame: Where a QUERY Record is the result of an ARP, ND, RARP, or unknown unicast MAC destination address, the ingress TRILL switch MAY send the frame to a Pull Directory Server if the frame is small enough that the resulting Query message fits into a TRILL Data packet within the campus MTU.

If no response is received to a Pull Directory Query message within a timeout configurable in milliseconds that defaults to 200, the Query message should be re-transmitted with the same Sequence Number up to a configurable number of times that defaults to three. If there are multiple QUERY Records in a Query message, responses can be received to various subsets of these QUERY Records before the timeout. In that case, the remaining unanswered QUERY Records should be re-sent in a new Query message with a new sequence number. If a TRILL switch is not capable of handling partial responses to queries with multiple QUERY Records, it MUST NOT sent a Request message with more than one QUERY Record in it.

See Section 3.5 for a discussion of how Query message errors are handled.

3.2.2 Pull Directory Response Format

Pull Directory Response messages are sent as the Channel Protocol specific content of an RBridge Channel message [RFC7178] TRILL Data packet or as a native RBridge Channel data frame (see Section 3.4). Responses are sent with the same Data Label and priority as the Query message to which they correspond except that the Response message priority is limited to be not more than a configured value. This priority limit is configurable at per TRILL switch and defaults to priority 6. Pull Directory Response messages SHOULD NOT be sent with priority 7 as that priority SHOULD be reserved for messages critical to network connectivity.

The RBridge Channel protocol specific data format is as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Ver  | Type  | Flags | Count |      Err      |    SubErr     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Sequence Number                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| RESPONSE 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
| RESPONSE 2
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
| ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
| RESPONSE K
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
```

Ver, Sequence Number: As specified in Section 3.1.

Type: 2 = Response.

Flags: MUST be sent as zero and ignored on receipt.

Count: Count is the number of RESPONSE Records present in the Response message.

Err, SubErr: A two part error code. Zero unless there was an error in the Query message, for which case see Section 3.5.

RESPONSE: Each RESPONSE record within a Pull Directory Response message is formatted as follows:

```
  0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|         SIZE          |OV|  RESV  |   Index   |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                   Lifetime                    |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                Response Data ...
+--+--+--+--+--+--+--+--+--+--+--...
```

SIZE: Size of the RESPONSE Record in bytes not counting the SIZE field and following byte. Thus the minimum value of SIZE is 2. If SIZE is less than 2, that RESPONSE Record and all subsequent RESPONSE Records in the Response message MUST be ignored and the entire Response message MAY be ignored.

OV: The overflow flag. Indicates, as described below, that there was too much Response Data to include in one Response message.

RESV: Three reserved bits that MUST be sent as zero and ignored on receipt.

Index: The relative index of the QUERY Record in the Query message to which this RESPONSE Record corresponds. The index will always be one for Query messages containing a single QUERY Record. If the Index is larger than the Count was in the corresponding Query, that RESPONSE Record MUST be ignored and subsequent RESPONSE Records or the entire Response message MAY be ignored.

Lifetime: The length of time for which the response should be considered valid in units of 200 milliseconds except that the values zero and 2**16-1 are special. If zero, the response can only be used for the particular query from which it resulted and MUST NOT be cached. If 2**16-1, the response MAY be kept indefinitely but not after the Pull Directory server goes down or becomes unreachable. The maximum definite time that can be expressed is a little over 3.6 hours.

Response Data: There are various types of RESPONSE Records.

- If the Err field is non-zero, then the Response Data is a copy of the corresponding QUERY Record data, that is, either an AFN followed by an address or a query frame. See Section 3.5 for additional information on errors.
- If the Err field is zero and the corresponding QUERY Record was an address query, then the Response Data is formated as the value of an Interface Addresses APPsub-TLV [IA]. The maximum size of such contents is 253 bytes in the case when SIZE is 255.

- If the Err field is zero and the corresponding QUERY Record was a frame query, then the Response data consists of the response frame for ARP, ND, or RARP and a copy of the frame for unknown unicast destination MAC.

Multiple RESPONSE Records can appear in a Response message with the same index if the answer to a QUERY Record consists of multiple Interface Address APPsub-TLV values. This would be necessary if, for example, a MAC address within a Data Label appears to be reachable by multiple TRILL switches. However, all RESPONSE Records to any particular QUERY Record MUST occur in the same Response message. If a Pull Directory holds more mappings for a queried address than will fit into one Response message, it selects which to include by some method outside the scope of this document and sets the overflow flag (OV) in all of the RESPONSE Records responding to that query address.

See Section 3.5 for a discussion of how errors are handled.

3.3 Cache Consistency

A Pull Directory MUST take action to minimize the amount of time that a TRILL switch will continue to use stale information from that Pull Directory by sending Update messages.

A Pull Directory server MUST maintain one of the following three sets of records, in order of increasing specificity. Retaining more specific records, such as that given in item 3 below, minimizes Spontaneous Update messages sent to update pull client TRILL switch caches but increases the record keeping burden on the Pull Directory server. Retaining less specific records, such as that given in item 1, will generally increase the volume and overhead due to Spontaneous Update messages and due to unnecessarily invalidating cached information, but will still maintain consistency and will reduce the record keeping burden on the Pull Directory server. In all cases, there may still be brief periods of time when directory information has changed but cached information a pull clients has not yet been updated or expunged.

1. An overall record per Data Label of when the last positive response data sent will expire at some requester and when the last negative response will expire at some requester, assuming those responders cached the response.

2. For each unit of data (IA APPsub-TLV Address Set [IA]) held by the server and each address about which ‘a negative response was sent, when the last response sent with that positive response data or negative response will expire at a requester, assuming the requester cached the response.

3. For each unit of data held by the server (IA APPsub-TLV Address Set [IA]) and each address about which a negative response was sent, a list of TRILL switches that were sent that data as a positive response or sent a negative response for the address, and the expected time to expiration for that data or address at each such TRILL switch, assuming the requester cached the response.

A Pull Directory server may have a limit as to how many TRILL switches for which it can maintain expiry information by method 3 above or how many data units or addresses it can maintain expiry information for by method 2. If such limits are exceeded, it MUST transition to a lower numbered strategy but, in all cases, MUST support, at a minimum, method 1.

When data at a Pull Directory changes or is deleted or data is added and there may be unexpired stale information at a requesting TRILL switch, the Pull Directory MUST send an Update message as discussed below. The sending of such an Update message MAY be delayed by a configurable number of milliseconds that default to 50 milliseconds to await other possible changes that could be included in the same Update.

If method 1, the most crude method, is being followed, then when any Pull Directory information in a Data Label is changed or deleted and there are outstanding cached positive data response(s), an all-addresses flush positive Update message is flooded within that Data Label as an RBridge Channel message with an Inner.MacDA of All-Egress-RBridges. And if data is added and there are outstanding cached negative responses, an all-addresses flush negative message is similarly flooded. "All-addresses" is indicated by the Count field being zero in an Update message. On receiving an all-addresses flooded flush positive Update from a Pull Directory server it has used, indicated by the F and P bits being one and the Count being zero, a TRILL switch discards all cached data responses it has for that Data Label. Similarly, on receiving an all addresses flush negative Update, indicated by the F and N bits being one and the Count being zero, it discards all cached negative replies for that Data Label. A combined flush positive and negative can be flooded by having all of the F, P, and N bits set to one resulting in the discard of all positive and negative cached information for the Data Label.

If method 2 is being followed, then a TRILL switch floods address specific positive Update messages when data that might be cached by a querying TRILL switch is changed or deleted and floods address specific negative Update messages when such information is added to. Such messages are similar to the method 1 flooded flush Update messages and are also sent as RBridge Channel messages with an Inner.MacDA of All-Egress-RBridges. However the Count field will be non-zero and either the P or N bit, but not both, will be one. On receiving such as address specific unsolicited update, if it is positive the addresses in the RESPONSE records in the unsolicited response are compared to the addresses about which the receiving TRILL switch is holding cached positive information from that server and, if they match, the cached information is updated. On receiving an address specific unsolicited update negative message, the addresses in the RESPONSE records in the unsolicited update are compared to the addresses about which the receiving TRILL switch is holding cached negative information from that server and, if they match, the cached negative information is updated.

If method 3 is being followed, the same sort of unsolicited update messages are sent as with method 2 above except they are not normally flooded but unicast only to the specific TRILL switches the directory server believes may be holding the cached positive or negative information that needs updating. However, a Pull Directory server MAY flood the unsolicited update under method 3, for example if it determines that a sufficiently large fraction of the TRILL switches in some Data label are requesters that need to be updated.

A Pull Directory server tracking cached information with method 3 MUST NOT clear the indication that it needs update cached information at a querying TRILL switch until it has sent an Update message and received a corresponding Acknowledge message or it has sent a configurable number of updates at a configurable interval which default to 3 updates 200 milliseconds apart.

A Pull Directory server tracking cached information with methods 2 or 1 SHOULD NOT clear the indication that it needs to update cached information until it has sent an Update message and received a corresponding Acknowledge message from all of its ESADI neighbors or it has sent a configurable number of updates at a configurable interval that defaults to 3 updates 200 milliseconds apart.

3.3.1 Update Message Format

An Update message is formatted as a Response message except that the Type field in the message header is a different value.

Update messages are initiated by a Pull Directory server. The Sequence number space used is controlled by the originating Pull Directory server and different from Sequence number space used in a Query and the corresponding Response that are controlled by the querying TRILL switch.

The Flags field of the message header for an Update message is as follows:

```
+---+---+---+---+
| F | P | N | R |
+---+---+---+---+
```

F: The Flood bit. If zero, the response is to be unicast . If F=1, it is multicast to All-Egress-RBridges.

P, N: Flags used to indicate positive or negative Update messages. P=1 indicates positive. N=1 indicates negative. Both may be 1 for a flooded all addresses Update.

R: Reserved. MUST be sent as zero and ignored on receipt

3.3.2 Acknowledge Message Format

An Acknowledge message is sent in response to an Update to confirm receipt or indicate an error unless response is inhibited by rate limiting. It is also formatted as a Response message.

If there are no errors in the processing of an Update message, the message is essentially echoed back with the Type changed to Acknowledge.

If there was an overall or header error in an Update message, it is echoed back as an Acknowledge message with the Err and SubErr fields set appropriately (see Section 3.5).

If there is a RESPONSE Record level error in an Update message, one or more Acknowledge messages may be returns as indicated in Section 3.5.

3.4 Pull Directory Hosted on an End Station

Optionally, a Pull Directory actually hosted on an end station MAY be supported. In that case, one or more TRILL switches must proxy for the end station and advertise themselves as a Pull Directory server. Such proxies must have a direct connection to the end station, that is a connection not involving any intermediate TRILL switches.

When the proxy TRILL switch receives a Query message, it modifies the inter-RBridge Channel message received into a native RBridge Channel message and forwards it to that end station. Later, when it receives one or more responses from that end station by native RBridge Channel messages, it modifies them into inter-RBridge Channel messages and forwards them to the source TRILL switch of the original Query message. Similarly, an Update from the end station is forwarded to client TRILL switches and acknowledgements from those TRILL switches are returned to the end station by the proxy. Because native RBridge Channel messages have no TRILL Header and are addressed by MAC address, as opposed to inter-RBridge Channel messages that are TRILL Data packets and are addressed by nickname, nickname information must be added to the native RBridge Channel version of Pull Directory messages.

The native Pull Directory RBridge Channel messages use the same Channel protocol number as do the inter-RBridge Pull Directory RBridge Channel messages. The native messages SHOULD be sent with an Outer.VLAN tag which gives the priority of each message which is the priority of the original inter-RBridge request packet. The Outer.VLAN ID used is the Designated VLAN on the link to the end station. Since there is no TRILL Header or inner Data Label for native RBridge Chanel messages, that information is added to the header.

The native RBridge Channel message Pull Directory message protocol dependent data part is the same as for inter-RBridge Channel messages except that the 8-byte header described in Section 3.1 is expanded to 14 or 18 bytes as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Ver  | Type  | Flags | Count |      Err      |    SubErr     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Sequence Number                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Nickname  (2 bytes)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...+-+
|   Data Label ... (4 or 8 bytes)                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...+-+
| Type Specific Payload - variable length
+-+-+- ...
```

Fields not described below are as in Section 3.1.

Data Label: The Data Label that normally appear right after the Inner.MacSA of the an RBridge Channel Pull Directory message appears here in the native RBridge Channel message version. This might appear in a Query message, to be reflected in a Response message, or it might appear in an Update message, to be reflected in an Acknowledge message.

Nickname: The nickname of the TRILL switch that is communicating with the end station Pull Directory. Usually this is a remote TRILL switch but it could be the TRILL switch to which the end station is attached. The proxy copies this from the ingress nickname when mapping a Query or Acknowledge message to native form. It also takes this from a native Response or Update to be used as the egress of the inter-RBridge form on the message unless it is a flooded Update in which case a distribution tree is used.

3.5 Pull Directory Message Errors

A non-zero Err field in the Pull Directory message header indicates an error message.

If there is an error that applies to an entire Query message or its header, as indicated by the range of the value of the Err field, then the QUERY records in the request are just echoed back in the RESPONSE records of the Response message but expanded with a zero Lifetime and the insertion of the Index field. If there is an error that applies to an entire Update message or its header, then the RESPONSE records in the update, if any, are echoed back in the Acknowledge message.

If errors occur at the QUERY Record level for a Query message, they MUST be reported in a Response message separate from the results of any successful non-erroneous QUERY Records. If multiple QUERY Records in a Query message have different errors, they MUST be reported in separate Response messages. If multiple QUERY Records in a Query message have the same error, this error response MAY be reported in one or multiple Response messages. In an error Response message, the QUERY Record or records being responded to appear, expanded by the Lifetime for which the server thinks the error might persist and with their Index inserted, as the RESPONSE record or records.

If errors occur at the RESPONSE Record level for an Update message, they MUST be reported in a Acknowledge message separate from the acknowledgement of any non-erroneous RESPONSE Records. If multiple RESPONSE Records in an Update have different errors, they MUST be reported in separate Acknowledge messages. If multiple RESPONSE Records in an Update message have the same error, this error response MAY be reported in one or multiple Acknowledge messages. In an error Acknowledge message, the RESPONSE Record or records being responded to appear, expanded by the time for which the server thinks the error might persist and with their Index inserted, as a RESPONSE Record or records.

ERR values 1 through 127 are available for encoding Request or Update message level errors. ERR values 128 through 254 are available for encoding QUERY or RESPONSE Record level errors. The SubErr field is available for providing more detail on errors. The meaning of a SubErr field value depends on the value of the Err field.

Err	Meaning
0	(no error)
1	Unknown or reserved Query message field value
2	Request data too short
3	Unknown or reserved Update message field value
4	Update data too short
5-127	(Available for allocation by IETF Review)
128	Unknown or reserved QUERY Record field value
129	Address not found
130	Unknown or reserved RESPONSE Record field value
131-254	(Available for allocation by IETF Review)
255	Reserved

The following sub-errors are specified under error code 1 and 3:

SubErr	Field with Error
0	Unspecified
1	Unknown V field value
2	Reserved T field value
3	Zero sequence number in request
4-254	(Available for allocation by Expert Review)
255	Reserved

The following sub-errors are specified under error code 128 and 130:

SubErr	Field with Error
0	Unspecified
1	Unknown AFN field value
2	Unknown or Reserved TYPE field value
3	Invalid or inconsistent SIZE field value
4-254	(Available for allocation by Expert Review)
255	Reserved

More TBD

3.6 Additional Pull Details

If a TRILL switch notices that a Pull Directory server is no longer data reachable [RFC7180], it MUST promptly discard all pull responses it is retaining from that server as it can no longer receive cache consistency update messages from the server.

Because a Pull Directory server may need to advertise interest in Data Labels even though it does not want to received end station data in those Data Labels, the No Data (NOD) flag bit is provided as specified in Section 6.3. For example, an RBridge hosting a Pull Directory may be a secondary directory that wants to receive its data from a primary Push Directory server but have no interest in receiving multicast traffic from end stations.

4. Directory Use Strategies and Push-Pull Hybrids

For some edge nodes that have a great number of Data Labels enabled, managing the MAC and Data Label <-> Edge RBridge mapping for hosts under all those Data Labels can be a challenge. This is especially true for Data Center gateway nodes, which need to communicate with a majority of Data Labels, if not all.

For those edge TRILL switch nodes, a hybrid model should be considered. That is the Push Model is used for some Data Labels, and the Pull Model is used for other Data Labels. It is the network operator’s decision by configuration as to which Data Labels’ mapping entries are pushed down from directories and which Data Labels’ mapping entries are pulled.

For example, assume a data center where hosts in specific Data Labels, say VLANs 1 through 100, communicate regularly with external peers. Probably, the mapping entries for those 100 VLANs should be pushed down to the data center gateway routers. For hosts in other Data Labels which only communicate with external peers occasionally for management interface, the mapping entries for those VLANs should be pulled down from directory when the need comes up.

The mechanisms described above for Push and Pull Directory services make it easy to use Push for some Data Labels and Pull for others. In fact, different TRILL switches can even be configured so that some use Push Directory services and some use Pull Directory services for the same Data Label if both Push and Pull Directory services are available for that Data Label. And there can be Data Labels for which directory services are not used at all.

For Data Labels in which a hybrid push/pull approach is being taken, it would make sense to use push for address information of hosts that frequently communicate with many other hosts in the Data Label, such as a file or DNS server. Pull could then be used for hosts that communicate with few other hosts, perhaps such as hosts being used as compute engines.

4.1 Strategy Configuration

Each TRILL switch that has the ability to use directory assistance has, for each Data Label X in which it is might ingress native frames, one of four major modes:

0. No directory use: The TRILL switch does not subscribe to Push Directory data or make Pull Directory requests for Data Label X and directory data is not consulted on ingressed frames in Data Label X that might have used directory data. This includes ARP,

ND, RARP, and unknown MAC destination addresses, which are flooded as appropriate.

1. Use Push only: The TRILL switch subscribes to Push Directory data for Data Label X.

2. Use Pull only: When the TRILL switch ingresses a frame in Data Label X that can use Directory information, if it has cached information for the address it uses it. If it does not have either cached positive or negative information for the address, it sends a Pull Directory query.

3. Use Push and Pull: The TRILL switch subscribes to Push Directory data for Data Label X. When it ingresses a frame in Data Label X that can use Directory information and it does not find that information in its link state database of Push Directory information, it makes a Pull Directory query.

The above major Directory use mode is per Data Label. In addition, there is a per Data Label per priority minor mode as listed below that indicates what should be done if Directory Data is not available for the ingressed frame. In all cases, if you are holding Push Directory or Pull Directory information to handle the frame given the major mode, the directory information is simply used and, in that instance, the minor mode does not matter.

A. Flood immediate: Flood the frame immediately (even if you are also sending a Pull Directory) request.

B. Flood: Flood the frame immediately unless you are going to do a Pull Directory request, in which case you wait for the response or for the request to time out after retries and flood the frame if the request times out.

C. Discard if complete or Flood immediate: If you have complete Push Directory information and the address is not in that information, discard the frame. If you do not have complete Push Directory information, the same as A above.

D. Discard if complete or Flood: If you have complete Push Directory information and the address is not in that information, discard the frame. If you do not have complete Push Directory information, the same as B above.

In addition, the query message priority for Pull Directory requests sent can be configured on a per Data Label, per ingressed frame priority basis. The default mappings are as follows where Ingress Priority is the priority of the native frame that provoked the Pull Directory query:

Ingress Priority	If Flood Immediate	If Flood Delayed
7	5	6
6	5	6
5	4	5
4	3	4
3	2	3
2	0	2
0	1	0
1	1	1

Priority 7 is normally only used for urgent messages critical to adjacency and so is avoided by default for directory traffic. Unsolicited updates are sent with a priority that is configured per Data Label that defaults to priority 5.

5. Security Considerations

Incorrect directory information can result in a variety of security threats including the following:

> Incorrect directory mappings can result in data being delivered to the wrong end stations, or set of end stations in the case of multi-destination packets, violation security policy.
>
> Missing or incorrect directory data can result in denial of service due to sending data packets to black holes or discarding data on ingress due to incorrect information that their destinations are not reachable.

Push Directory data is distributed through ESADI-LSPs [RFC7357] that can be authenticated with the same mechanisms as IS-IS LSPs. See [RFC5304] [RFC5310] and the Security Considerations section of [RFC7357].

Pull Directory queries and responses are transmitted as RBridge-to-RBridge or native RBridge Channel messages. Such messages can be secured as specified in [ChannelTunnel].

For general TRILL security considerations, see [RFC6325].

6. IANA Considerations

This section gives IANA assignment and registry considerations.

6.1 ESADI-Parameter Data Extensions

IANA will assigned two ESADI-Parameter TRILL APPsub-TLV flag bits for "Push Directory" (PSH) and "Complete Push" (COP) and will create a sub-registry in the TRILL Parameters Registry as follows:

Sub-Registry: ESADI-Parameter APPsub-TLV Flag Bits

Registration Procedures: Standards Action

References: [RFC7357] [This document]

Bit	Mnemonic	Description	Reference
0	UN	Supports Unicast ESADI	ESDADI [RFC7357]
1	PSH	Push Directory Server	This document
2	COP	Complete Push	This document
3-7	-	available for allocation	

The COP bit is ignored if the PSH bit is zero.

In addition, the ESADI-Parameter APPsub-TLV is optionally extended, as provided in its original specification in ESDADI [RFC7357], by one byte as show below:

```
+-+-+-+-+-+-+-+-+
| Type          |            (1 byte)
+-+-+-+-+-+-+-+-+
| Length        |            (1 byte)
+-+-+-+-+-+-+-+-+
|R| Priority    |            (1 byte)
+-+-+-+-+-+-+-+-+
| CSNP Time     |            (1 byte)
+-+-+-+-+-+-+-+-+
| Flags         |            (1 byte)
+---------------+
|PushDirPriority|            (optional, 1 byte)
+---------------+
| Reserved for expansion     (variable)
+-+-+-+-...
```

The meanings of all the fields are as specified in ESDADI [RFC7357] except that the added PushDirPriority is the priority of the advertising ESADI instance to be a Push Directory as described in

Section 2.3. If the PushDirPriority field is not present (Length = 3) it is treated as if it were 0x40. 0x40 is also the value used and placed here by an TRILL switch whose priority to be a Push Directory has not been configured.

6.2 RBridge Channel Protocol Number

IANA will allocate a new RBridge Channel protocol number for "Pull Directory Services" from the range allocable by Standards Action and update the subregistry of such protocol number in the TRILL Parameters Registry referencing this document.

6.3 The Pull Directory (PUL) and No Data (NOD) Bits

IANA is requested to allocate two currently reserved bits in the Interested VLANs field of the Interested VLANs sub-TLV (suggested bits 18 and 19) and the Interested Labels field of the Interested Labels sub-TLV (suggested bits 6 and 7) [RFC7176] to indicate Pull Directory server (PUL) and No Data (NOD) respectively. These bits are to be added, with this document as reference, to the "Interested VLANs Flag Bits" and "Interested Labels Flag Bits" subregistries created by [RFC7357].

{{Material below in this subsection is technical and should be moved out of the IANA Consdierations.}}

In the TRILL base protocol [RFC6325] as extended for FGL [RFC7172], the mere presence of an Interested VLANs or Interested Labels sub-TLVs in the LSP of a TRILL switch indicates connection to end stations in the VLAN(s) or FGL(s) listed and thus a desire to receive multi-destination traffic in those Data Labels. But, with Push and Pull Directories, advertising that you are a directory server requires using these sub-TLVs to indicate the Data Label(s) you are serving. If such a directory server does not wish to received multi-destination TRILL Data packets for the Data Labels it lists in one of these sub-TLVs, it sets the "No Data" (NOD) bit to one. This means that data on a distribution tree may be pruned so as not to reach the "No Data" TRILL switch as long as there are no TRILL switches interested in the Data that are beyond the "No Data" TRILL switch on a distribution tree. The NOD bit is backwards compatible as TRILL switches ignorant of it will simply not prune when they could, which is safe although it may cause increased link utilization.

Example of a TRILL switch serving as a directory that might not want multi-destination traffic in some Data Labels would be a TRILL switch that does not offer end station service for any of the Data Labels

for which it is serving as a directory and is either
- a Pull Directory and/or
- a Push Directory for which all of the ESADI traffic will be handled by unicast ESDADI [RFC7357].

A Push Directory MUST NOT set the NOD bit for a data label if it needs to communicate via multi-destination ESADI PDUs in that data label since such PDUs look like TRILL Data packets to transit TRILL switches and might be incorrectly pruned if NOD was set.

Acknowledgments

The contributions of the following persons are gratefully acknowledged:

TBD

The document was prepared in raw nroff. All macros used were defined within the source file.

Normative References

[RFC826] - Plummer, D., "An Ethernet Address Resolution Protocol", RFC 826, November 1982.

[RFC903] - Finlayson, R., Mann, T., Mogul, J., and M. Theimer, "A Reverse Address Resolution Protocol", STD 38, RFC 903, June 1984

[RFC2119] - Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997

[RFC3971] - Arkko, J., Ed., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, March 2005.

[RFC4861] - Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.

[RFC5304] Li, T. and R. Atkinson, "IS-IS Cryptographic Authentication", RFC 5304, October 2008.

[RFC5310] - Bhatia, M., Manral, V., Li, T., Atkinson, R., White, R., and M. Fanto, "IS-IS Generic Cryptographic Authentication", RFC 5310, February 2009.

[RFC6165] - Banerjee, A. and D. Ward, "Extensions to IS-IS for Layer-2 Systems", RFC 6165, April 2011.

[RFC6325] - Perlman, R., Eastlake 3rd, D., Dutt, D., Gai, S., and A. Ghanwani, "Routing Bridges (RBridges): Base Protocol Specification", RFC 6325, July 2011.

[RFC7042] - Eastlake 3rd, D. and J. Abley, "IANA Considerations and IETF Protocol and Documentation Usage for IEEE 802 Parameters", BCP 141, RFC 7042, October 2013.

[RFC7172] - Eastlake 3rd, D., Zhang, M., Agarwal, P., Perlman, R., and D. Dutt, "Transparent Interconnection of Lots of Links (TRILL): Fine-Grained Labeling", RFC 7172, May 2014, <http://www.rfc-editor.org/info/rfc7172>.

[RFC7176] - Eastlake 3rd, D., Senevirathne, T., Ghanwani, A., Dutt, D., and A. Banerjee, "Transparent Interconnection of Lots of Links (TRILL) Use of IS-IS", RFC 7176, May 2014, <http://www.rfc-editor.org/info/rfc7176>.

[RFC7178] - Eastlake 3rd, D., Manral, V., Li, Y., Aldrin, S., and D. Ward, "Transparent Interconnection of Lots of Links (TRILL): RBridge Channel Support", RFC 7178, May 2014, <http://www.rfc-

editor.org/info/rfc7178>.

[RFC7180] - Eastlake 3rd, D., Zhang, M., Ghanwani, A., Manral, V., and A. Banerjee, "Transparent Interconnection of Lots of Links (TRILL): Clarifications, Corrections, and Updates", RFC 7180, May 2014, <http://www.rfc-editor.org/info/rfc7180>.

[RFC7357] - Zhai, H., Hu, F., Perlman, R., Eastlake 3rd, D., and O. Stokes, "Transparent Interconnection of Lots of Links (TRILL): End Station Address Distribution Information (ESADI) Protocol", RFC 7357, September 2014, <http://www.rfc-editor.org/info/rfc7357>.

[IA] - Eastlake, D., L. Yizhou, R. Perlman, "TRILL: Interface Addresses APPsub-TLV", draft-eastlake-trill-ia-appsubtlv, work in progress.

Informational References

[RFC7067] - Dunbar, L., Eastlake 3rd, D., Perlman, R., and I. Gashinsky, "Directory Assistance Problem and High-Level Design Proposal", RFC 7067, November 2013.

[ChannelTunnel] - D. Eastlake, Y. Li, "TRILL: RBridge Channel Tunnel Protocol", draft-eastlake-trill-channel-tunnel, work in progress.

[ARP reduction] - Shah, et. al., "ARP Broadcast Reduction for Large Data Centers", Oct 2010.

Authors' Addresses

Linda Dunbar
Huawei Technologies
5430 Legacy Drive, Suite #175
Plano, TX 75024, USA

Phone: +1-469-277-5840
Email: ldunbar@huawei.com

Donald Eastlake
Huawei Technologies
155 Beaver Street
Milford, MA 01757 USA

Phone: +1-508-333-2270
Email: d3e3e3@gmail.com

Radia Perlman
EMC
2010 256th Avenue NE, #200
Bellevue, WA 98007 USA

Email: Radia@alum.mit.edu

Igor Gashinsky
Yahoo
45 West 18th Street 6th floor
New York, NY 10011

Email: igor@yahoo-inc.com

Yizhou Li
Huawei Technologies
101 Software Avenue,
Nanjing 210012 China

Phone: +86-25-56622310
Email: liyizhou@huawei.com

TRILL working group
Internet Draft
Intended status: Standard Track
Expires: June 2015
L. Dunbar
D. Eastlake
Huawei
Radia Perlman
Intel
I. Gashinsky
Yahoo
December 16, 2014

Directory Assisted TRILL Encapsulation
draft-ietf-trill-directory-assisted-encap-00.txt

Status of this Memo

Copyright Notice

Abstract

This draft describes how data center network can benefit from non-RBridge nodes performing TRILL encapsulation with assistance from directory service.

Table of Contents

1. Introduction

This draft describes how data center networks can benefit from non-RBridge nodes performing TRILL encapsulation with assistance from directory service.

[RFC7067] describes the framework for RBridge edge to get MAC&VLAN<->RBridgeEdge mapping from a directory service in data center environments instead of flooding unknown DAs across TRILL domain. If it has the needed directory information, any node, even a non-RBridge node, can perform the TRILL encapsulation. This draft is to describe the benefits and a scheme for non-RBridge nodes performing TRILL encapsulation.

2. Conventions used in this document

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 RFC-2119 [RFC2119].

In this document, these words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying RFC-2119 significance.

AF Appointed Forwarder RBridge port [RFC6439]

Bridge: IEEE 802.1Q compliant device. In this draft, Bridge is used interchangeably with Layer 2 switch.

DA: Destination Address

DC: Data Center

EoR: End of Row switches in data center. Also known as Aggregation switches in some data centers

Host: Application running on a physical server or a virtual machine. A host usually has at least one IP address and at least one MAC address.

SA: Source Address

ToR: Top of Rack Switch in data center. It is also known as access switches in some data centers.

TRILL-EN: TRILL Encapsulating node. It is a node that only performs the TRILL encapsulation but doesn't participate in RBridge's IS-IS routing.

VM: Virtual Machines

3. Directory Assistance to Non-RBridge

With directory assistance [RFC7067], a non-RBridge can be informed if a packet needs to be forwarded across the RBridge domain and the corresponding egress RBridge. Suppose the RBridge domain boundary starts at network switches (not virtual switches embedded on servers), a directory can assist Virtual Switches embedded on servers to encapsulate with a proper TRILL header by providing the nickname of the egress RBridge edge to which the destination is attached. The other information needed to encapsulate can be either learned by listening to TRILL Hellos, which will indicate the MAC address and nickname of appropriate edge RBridges, or by configuration.

If a destination is not attached to other RBridge edge nodes based on the directory [RFC7067], the non-RBridge node can forward the data frames natively, i.e. not encapsulating any TRILL header.

```
      \             +-------+         +------+ TRILL Domain/
       \          +/------+ |       +/-----+ |           /
        \         | Aggr11| + ----- |AggrN1| +          /
         \        +---+---+/        +------+/          /
          \         /     \           /      \        /
           \      /        \        /         \      /
            \  +---+    +---+      +---+     +---+   /
             \- |T11|... |T1x|      |T21| ..  |T2y|---
               +---+    +---+      +---+     +---+
                 |        |          |         |
               +-|-+    +-|-+      +-|-+     +-|-+
               |   |... | V |      | V | ..  | V |<- vSwitch
               +---+    +---+      +---+     +---+
               |   |... | V |      | V | ..  | V |
               +---+    +---+      +---+     +---+
               |   |... | V |      | V | ..  | V |
               +---+    +---+      +---+     +---+
```

Figure 1 TRILL domain in typical Data Center Network

When a TRILL encapsulated data packet reaches the ingress RBridge, the ingress RBridge simply forwards the pre-encapsulated packet to the RBridge that is specified by the egress nickname field of the TRILL header of the data frame. When the ingress RBridge receives a native Ethernet frame, it handles it as usual and may drop it if it has complete directory information indicating that the target is not attached to the TRILL campus.

In this environment with complete directory information, the ingress RBridge doesn’t flood or forward the received data frames when the DA in the Ethernet data frames is unknown.

When all attached nodes to ingress RBridge can pre-encapsulate TRILL header for traffic across the TRILL domain, the ingress RBridge don’t need to encapsulate any native Ethernet frames to the TRILL domain. The attached nodes can be connected to multiple edge RBridges by having multiple ports or by an bridged LAN. Under this environment, there is no need to designate AF ports and all RBridge edge ports connected to one bridged LAN can receive and forward pre-encapsulated traffic, which can greatly improve the overall network utilization.

Note: [RFC6325] Section 4.6.2 Bullet 8 specifies that an RBridge port can be configured to accept TRILL encapsulated frames from a neighbor that is not an RBridge.

When a TRILL frame arrives at an RBridge whose nickname matches with the destination nickname in the TRILL header of the frame, the processing is exactly same as normal, i.e. the RBridge decapsulates the received TRILL frame and forwards the decapsulated frame to the target attached to its edge ports. When the DA of the decapsulated Ethernet frame is not in the egress RBridge's local MAC attachment tables, the egress RBridge floods the decapsulated frame to all attached links in the frame's VLAN, or drops the frame (if the egress RBridge is configured with the policy).

We call a node that only performs the TRILL encapsulation but doesn't participate in RBridge's IS-IS routing a TRILL Encapsulating node (TRILL-EN). The TRILL Encapsulating Node can get the MAC&VLAN<->RBridgeEdge mapping table pulled from directory servers [RFC7067].

Editor's note: RFC7067 has defined Push and Pull model for edge nodes to get directory mapping information. While Pull Model is relative simple for TRILL-EN to implement, Pushing requires some reliable flooding mechanism, like the one used by IS-IS, between the edge RBridge and the TRILL encapsulating node. Something like an extension to ES-IS might be needed.

Upon receiving a native Ethernet frame, the TRILL-EN checks the MAC&VLAN<->RBridgeEdge mapping table, and perform the corresponding TRILL encapsulation if the entry is found in the mapping table. If the destination address and VLAN of the received Ethernet frame doesn't exist in the mapping table and no positive reply from pulling request to a directory, the Ethernet frame is dropped or forwarded in native form to an edge RBridge.

```
       +------------+--------+---------+---------+--+-------+---+
       |OuterEtherHd|TRILL HD| InnerDA | InnerSA |..|Payload|FCS|
       +------------+--------+---------+---------+--+-------+---+
               ^
               |             |<Inner Ether Header>  |
               |
               |
               |      +-------+  TRILL    +------+
               |      |  R1   |-----------|  R2  |  Decapsulate
               |      +---+---+  domain   +------+  TRILL header
               |          |                   |
               +----------|                   |
                          |                   |
                       +-----+             +-----+
      Non-RBridge node:|T12  |             | T22 |
      Encapsulate TRILL+-----+             +-----+
      Header for data
      Frames to traverse
      TRILL domain.
```

Figure 2 Data frames from TRILL-EN

4. Source Nickname in Frames Encapsulated by Non-RBridge Nodes

The TRILL header includes a Source RBridge’s Nickname (ingress) and Destination RBridge’s Nickname (egress). When a TRILL header is added by TRILL-EN, the Ingress RBridge edge node’s nickname is used in the source address field.

5. Benefits of Non-RBridge encapsulating TRILL header

5.1. Avoid Nickname Exhaustion Issue

For a large Data Center with hundreds of thousands of virtualized servers, setting the TRILL boundary at the servers’ virtual switches will create a TRILL domain with hundreds of thousands of RBridge nodes, which has issues of TRILL Nicknames exhaustion and challenges to IS-IS. On the other hand, setting TRILL boundary at aggregation switches that have many virtualized servers attached can limit the number of RBridge nodes in a TRILL domain, but introduce the issues of very large MAC&VLAN<->RBridgeEdge mapping table to be maintained by RBridge edge nodes and the necessity of enforcing AF ports.

Allowing Non-RBridge nodes to pre-encapsulate data frames with TRILL header makes it possible to have a TRILL domain with a reasonable number of RBridge nodes in a large data center. All the TRILL-ENs attached to one RBridge are represented by one TRILL nickname, which can avoid the Nickname exhaustion problem.

5.2. Reduce MAC Tables for switches on Bridged LANs

When hosts in a VLAN (or subnet) span across multiple RBridge edge nodes and each RBridge edge has multiple VLANs enabled, the switches on the bridged LANs attached to the RBridge edge are exposed to all MAC addresses among all the VLANs enabled.

For example, for an Access switch with 40 physical servers attached, where each server has 100 VMs, there are 4000 hosts under the Access Switch. If indeed hosts/VMs can be moved anywhere, the worst case for the Access Switch is when all those 4000 VMs belong to different VLANs, i.e. the access switch has 4000 VLANs enabled. If each VLAN has 200 hosts, this access switch’s MAC table potentially has 200*4000 = 800,000 entries.

If the virtual switches on servers pre-encapsulate the data frames destined for hosts attached to other RBridge Edge nodes, the outer MAC DA of those TRILL encapsulated data frames will be the MAC address of the local RBridge edge, i.e. the ingress RBridge. Therefore, the switches on the local bridged LAN don’t need to keep the MAC entries for remote hosts attached to other edge RBridges.

But the traffic from nodes attached to other RBridges is decapsulated and has the true source and destination MACs. To prevent local bridges from learning remote hosts’ MACs and adding to their MAC tables, one simple way is to disable this data plane learning on local bridges. The local bridges can be pre-configured with MAC addresses of local hosts with the assistance of a directory. The local bridges can always send frames with unknown Destination to the ingress RBridge. In an environment where a large number of VMs are instantiated in one server, the number of remote MAC addresses could be very large. If it is not feasible to disable learning and pre-configure MAC tables for local bridges, one effective method to minimize local bridges’ MAC table size is to use the server’s MAC address to hide MAC addresses of the attached VMs. I.e. the server acting as an edge node using its own MAC address in the Source Address field of the packets originated from a host (or VM) embedded. When the Ethernet frame arrives at the target edge node (the server), the target edge node can send the packet to the corresponding destination host based on the packet’s IP address. Very often, the target edge node communicates with the embedded VMs via a layer 2 virtual switch. Under this case, the target edge node can construct the proper Ethernet header with the assistance from directory. The information from directory includes the proper host IP to MAC mapping information.

6. Conclusion and Recommendation

When directory information is available, nodes outside the TRILL domain can encapsulate data frames destined for nodes attached to remote RBridges. The non-RBridge encapsulation approach is especially useful when there are a large number of servers in a data center equipped with hypervisor-based virtual switches. It is relatively easy for virtual switches, which are usually software based, to get directory assistance and perform network address encapsulation.

7. Manageability Considerations

It requires directory assistance to make it possible for a non-TRILL node to pre-encapsulate packets destined towards remote RBridges.

8. Security Considerations

Pull Directory queries and responses are transmitted as RBridge-to-RBridge or native RBridge Channel messages. Such messages can besecured as specified in [ChannelTunnel].

For general TRILL security considerations, see [RFC6325].

9. IANA Considerations

This document requires no IANA actions. RFC Editor: Please remove this section before publication.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC6325] Perlman, et, al, "Routing Bridges (RBridges): Base Protocol Specification", RFC6325, July 2011

[RFC6439] Perlman, R., Eastlake, D., Li, Y., Banerjee, A., and F. Hu, "Routing Bridges (RBridges): Appointed Forwarders", RFC 6439, November 2011.

10.2. Informative References

[RFC7067] Dunbar, et, al "Directory Assistance Problem and High-Level Design Proposal", RFC7067, Nov, 2013.

[ChannelTunnel] - D. Eastlake, Y. Li, "TRILL: RBridge Channel Tunnel Protocol", draft-eastlake-trill-channel-tunnel, work in progress.

11. Acknowledgments

This document was prepared using 2-Word-v2.0.template.dot.

Authors' Addresses

Linda Dunbar
Huawei Technologies
5340 Legacy Drive, Suite 175
Plano, TX 75024, USA
Phone: (469) 277 5840
Email: linda.dunbar@huawei.com

Donald Eastlake
Huawei Technologies
155 Beaver Street
Milford, MA 01757 USA
Phone: 1-508-333-2270
Email: d3e3e3@gmail.com

Radia Perlman
Intel Labs
2200 Mission College Blvd.
Santa Clara, CA 95054-1549 USA
Phone: 1-408-765-8080
Email: Radia@alum.mit.edu

Igor Gashinsky
Yahoo
45 West 18th Street 6th floor
New York, NY 10011
Email: igor@yahoo-inc.com

INTERNET-DRAFT   Donald Eastlake
Intended status: Proposed Standard   Yizhou Li
                                                                   Huawei
                                                             Radia Perlman
                                                                      EMC
Expires: May 23, 2014   November 24, 2014

TRILL: Interface Addresses APPsub-TLV
<draft-ietf-trill-ia-appsubtlv-02.txt>

Abstract

This document specifies a TRILL (Transparent Interconnection of Lots of Links) IS-IS application sub-TLV that enables the reporting by a TRILL switch of sets of addresses such that all of the addresses in each set designate the same interface (port) and the reporting for such a set of the TRILL switch by which it is reachable. For example, a 48-bit MAC (Media Access Control) address, IPv4 address, and IPv6 address can be reported as all corresponding to the same interface reachable by a particular TRILL switch. Such information could be used in some cases to synthesize responses to or by-pass the need for the Address Resolution Protocol (ARP), the IPv6 Neighbor Discovery (ND) protocol, or the flooding of unknown MAC addresses.

Status of This Memo

Table of Contents

1. Introduction

This document specifies a TRILL (Transparent Interconnection of Lots of Links) [RFC6325] IS-IS application sub-TLV (APPsub-TLV [RFC6823]) that enables the convenient representation of sets of addresses such that all of the addresses in each set designate the same interface (port). For example, a 48-bit MAC (Media Access Control [RFC7042]) address, IPv4 address, and IPv6 address can be reported as all three designating the same interface. In addition, a Data Label (VLAN or Fine Grained Label (FGL [RFC7172])) is specified for the interface along with the TRILL switch, and optionally the TRILL switch port, from which the interface is reachable. Such information could be used in some cases to synthesize responses to or by-pass the need for the Address Resolution Protocol (ARP [RFC826]), the IPv6 Neighbor Discovery (ND [RFC4861]) protocol, the Reverse Address Resolution Protocol (RARP [RFC903]), or the flooding of unknown destination MAC addresses [RFC7042]. If the information report is complete, it can also be used to detect and discard packets with forged source addresses.

This APPsub-TLV appears inside the TRILL GENINFO TLV specified in ESADI [RFC7357] but may also occur in other application contexts. Directory Assisted TRILL Edge services [DirectoryScheme] are expected to make use of this APPsub-TLV.

Although, in some IETF protocols, address field types are represented by Ethertype [RFC7042] or Hardware Type [RFC5494], only Address Family Number (AFN) is used in this APPsub-TLV to represent address field type.

1.1 Conventions Used in This Document

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]. Capitalized IANA Considertions terms such as "Expert Review" are to be interpreted as described in [RFC5226].

The terminology and acronyms of [RFC6325] are used herein along with the following additional acronyms and terms:

AFN: Address Family Number

APPsub-TLV: Application sub-TLV [RFC6823]

Data Label: VLAN or FGL

FGL: Fine Grained Label [RFC7172]

IA: Interface Addresses

RBridge: An alternative name for a TRILL switch

TRILL switch: A device that implements the TRILL protocol

2. Format of the Interface Addresses APPsub-TLV

The Interface Addresses (IA) APPsub-TLV is used to advertise that a set of addresses indicate the same interface (port) within a Data Label (VLAN or FGL) and to associate that interface with the TRILL switch, and optionally the TRILL switch port, by which the interface is reachable. These addresses can be in different address families. For example, it can be used to declare that a particular interface with specified IPv4, IPv6, and 48-bit MAC addresses in some particular Data Label is reachable from a particular TRILL switch.

The Template field in a particular Interface Addresses APPsub-TLV indicates the format of each Address Set it carries. Certain well-known sets of addresses are represented by special values. Other sets of addresses are specified by a list of AFNs. The Template format that uses a list of AFNs provides an explicit pattern for the type and order of addresses in each Address Set in the IA APPsub-TLV that includes that Template.

A device or application making use of IA APPsub-TLV data is not required to make use of all IA data. For example, a device or application that was only interested in MAC and IPv6 addresses could ignore any IPv4 or other types of address information that was present.

The figure below shows an IA APPsub-TLV as it would appear inside an IS-IS FS-LSP using an extended flooding scope [RFC7356] TLV, for example in ESADI [RFC7357]. Within an IS-IS PDU using traditional [ISO-10589] TLVs, the Type and Length would be one byte unsigned integers equal to or less than 255.

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Type = TBD1                   |  (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Length                        |  (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Addr Sets End                 |  (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Nickname                      |  (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Flags         |                  (1 byte)
+-+-+-+-+-+-+-+-+
| Confidence    |                  (1 byte)
+-+-+-+-+-+-+-+-+-+-
| Template ...                     (variable)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...-+
| Address Set 1    (size determined by Template)    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...-+
| Address Set 2    (size determined by Template)    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...-+
|   ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...-+
| Address Set N    (size determined by Template)    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...-+
| optional sub-sub-TLVs ...
+-+-+-+-+-+-+-+-+-+-+-+-+-...
```

Figure 1. The Interface Addresses APPsub-TLV

o Type: Interface Addresses TRILL APPsub-TLV type, set to TBD1 (IA-SUBTLV).

o Length: Variable, minimum 7. If length is 6 or less or if the APPsub-TLV extends beyond the size of an encompassing TRILL GENINFO TLV or other context, the APPsub-TLV MUST be ignored.

o Addr Sets End: The unsigned integer offset of the byte, within the IA APPsub-TLV value part, of the last byte of the last Address Set. This will be the byte just before the first sub-sub-TLV if any sub-sub-TLVs are present (see Section 3). If this is equal to Length, there are no sub-sub-TLVs. If this is greater than Length or points to before the end of the Template, the IA APPsub-TLV is corrupt and MUST be discarded. This field is always two bytes in size.

o Nickname: The nickname of the TRILL switch by which the address sets are reachable. If zero, the address sets are reachable from the TRILL switch originating the message containing the APPsub-TLV (for example, an ESADI [RFC7357] message).

o Flags: A byte of flags as follows:

```
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|D|L|N|  RESV   |
+-+-+-+-+-+-+-+-+
```

D: Directory flag: If D is one, the APPsub-TLV contains Directory information [RFC7067].

L: Local flag: If L is one, the APPsub-TLV contains information learned locally by observing ingressed frames [RFC6325]. (Both D and L can be one in the same IA APPsub-TLV if a TRILL switch that had learned an address locally and also advertised it as a directory.)

N: Notify flag: When a TRILL switch receives a new IA APPsub-TLV (one in a ESADI-LSP fragment with a higher sequence number or a new message of some other type) and the N bit is one, the TRILL switch then checks the contents of the APPsub-TLV for address sets including both an IP address and a MAC address. For each such address set it finds, a gratuitous ARP [RFC826] or spontaneous Neighbor Advertisement [RFC4861], depending on whether the IP address is IPv4 or IPv6 respectively, may be sent. In both cases, these are sent out all the ports of the TRILL switch offering end station service and are in the VLAN or FGL of the address set information, that is, are Appointed Forwarder for the VLAN or for the VLAN to which the FGL maps.

RESV: Additional reserved flag bits that MUST be sent as zero and ignored on receipt.

o Confidence: This 8-bit unsigned quantity in the range 0 to 254 indicates the confidence level in the addresses being transported [RFC6325]. A value of 255 is treated as if it was 254.

o Template: The initial byte of this field is the unsigned integer K. If K has a value from 1 to 31, it indicates that this initial byte is followed by a list of K AFNs (Address Family Numbers) that specify the exact structure and order of each Address Set occurring later in the APPsub-TLV. K can be 1, which is the minimum valid value. If K is zero, the IA APPsub-TLV is ignored. If K is 32 to 254, the length of the Template field is one byte and its value is intended to correspond to a particular ordered set of AFNs some of which are specified below. If K is 255, the length of the Template filed is three bytes and the values of the second and third byte, considered as an unsigned integer in network byte order, are reserved to correspond to future specified ordered sets of AFNs.

If the Template uses explicit AFNs, it looks like the following, with the number of AFNs up to 31 equal to K.

```
+-+-+-+-+-+-+-+-+
|  K            |                 (1 byte)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  AFN 1                          |  (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  AFN 2                          |  (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  AFN K                          |  (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

For K in the 32 to 102 range, values indicate combinations of a specific number of MAC addresses, IPv4 addresses, IPv6 addresses, and TRILL switch port IDs appearing in that order. The value of K is

K = 31 + M + 3*v4 + 9*v6 + 36*P

where M is 0, 1, or 2 (0 if no MAC address is present, 1 if a 48-bit MAC is present, 2 if a MAC/24 (see Section 5.1) is present), v4 is the number of IPv4 addresses (limited to 0, 1, or 2) and v6 is the number of IPv6 addresses (limited to 0 through 3 inclusive), and P is the number of TRILL switch port IDs (limited to 0 or 1); however, the number of MAC, IPv4, and IPv6 addresses and TRILL switch ports cannot all be simultaneously zero. That equation specifies values of K from 32 through 102, the value 31 not being permitted but instead representing an explicit Template with 31 AFNs. Values from 103 through 254 of the byte value are available for assignment by Expert Review (see Section 5). K = 255 indicates a three-byte Template field as specified above. All values (0 through 65,545) of this two-byte value are available for assignment by Expert Review.

If an unknown Template K value in the range 103 to 254 is received or a K of 255 followed by an unknown two byte value, the IA APPsub-TLV MUST be ignored.

- AFN: A two-byte Address Family Number. The number of AFNs present is given by K except that there are no AFNs if K is greater than 31. The AFN sequence specifies the structure of the Address Sets occurring later in the TLV. For example, if Template Size is 2 and the two AFNs present are the AFNs for a 48-bit MAC and an IPv4 address, in that order, then each Address set present will consist of a 6-byte MAC address followed by a 4-byte IPv4 address. If any AFNs are present that are unknown to the receiving IS and the length of the corresponding address is not provided by a sub-sub-

TLV as specified below, the receiving IS will be unable to parse the Address Sets and MUST ignore the IA APPsub-TLV.

- Address Set: Each address set in the APPsub-TLV consists of exactly the same sequence of addresses of the types specified by the Template earlier in the APPsub-TLV. No alignment, other than to a byte boundary, is guaranteed. The addresses in each Address Set are contiguous with no unused bytes between them and the Address Sets are contiguous with no unused bytes between successive Address Sets. The Address Sets must fit within the TLV.

- sub-sub-TLVs: If the Address Sets indicated by Addr Sets End do not completely fill the Length of the APPsub-TLV, the remaining bytes are parsed as sub-sub-TLVs [RFC5305]. Any such sub-sub-TLVs that are not known to the receiving TRILL switch are ignored. Should this parsing not be possible, for example there is only one remaining byte or an apparent sub-sub-TLV extends beyond the end of the TLV, the containing IA APPsub-TLV is considered corrupt and is ignored. (Several sub-sub-TLV types are specified in Section 3.)

Different IA APPsub-TLVs within the same or different LSPs or other data structures may have different Templates. The same AFN may occur more than once in a Template and the same address may occur in different address sets. For example, a 48-bit MAC address interface might have three different IPv6 addresses. This could be represented by an IA APPsub-TLV whose Template specifically provided for one EUI-48 address and three IPv6 addresses, which might be an efficient format if there were multiple interfaces with that pattern. Alternatively, a Template with one 48-bit MAC and one IPv6 address could be used in an IA APPsub-TLV with three address sets each having the same MAC address but different IPv6 addresses, which might be the most efficient format if only one interface had multiple IPv6 addresses and other interfaces had only one IPv6 address.

In order to be able to parse the Address Sets, a receiving TRILL switch must know at least the size of the address for each AFN or address type the Template specifies; however, the presence of the Addr Set End field means that the sub-sub-TLVs, if any, can always be located by a receiver. A TRILL switch can be assumed to know the size of the AFNs mentioned in Section 5. Should a TRILL switch wish to include an AFN that some receiving TRILL switch in the campus may not know, it SHOULD include an AFN-Size sub-sub-TLV as described in Section 3.1. If an IA APPsub-TLV is received with one or more AFNs in its template for which the receiving TRILL switch does not know the length and for which an AFN-Size sub-sub-TLV is not present, that IA APPsub-TLV MUST be ignored.

3. IA APPsub-TLV sub-sub-TLVs

IA APPsub-TLVs can have trailing sub-sub-TLVs [RFC5305] as specified below. These sub-sub-TLVs occur after the Address Sets and the amount of space available for sub-sub-TLVs is determined from the overall IA APPsub-TLV length and the value of the Addr Set End byte.

There is no ordering restriction on sub-sub-TLVs. Unless otherwise specified each sub-sub-TLV type can occur zero, one, or many times in an IA APPsub-TLV. Any sub-sub-TLVs for which the Type is unknown are ignored.

The sub-sub-TLVs data structures shown below, with two byte Types and Lengths, assume that the enclosing IA-APPsubTLV is in an extended LSP TLV [RFC7356] or some non-LSP context. If they were used in a IA-APPsubTLV in a traditional LSP [ISO-10589], the only one byte Types and Lengths could be used. As a result, any sub-sub-TLV types greater than 255 could not be used and Length would be limited to 255.

3.1 AFN Size sub-sub-TLV

Using this sub-sub-TLV, the originating TRILL switch can specify the size of an address type. This is useful under two circumstances as follows:

1. One or more AFNs that are unknown to the receiving TRILL switch appears in the template. If an AFN Size sub-sub-TLV is present for each such AFN, then at least the IA APPsub-TLV can be parsed and possibly other addresses in each address set can still be used.

2. If an AFN occurs in the Template that represents a variable length address, this sub-sub-TLV gives its size for all occurrences in that IA APPsub-TLV.

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Type = AFNsz                  |  (2 byte)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Length                        |  (2 byte)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| AFN Size Record 1                                             |  (3 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| AFN Size Record 2                                             |  (3 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| AFN Size Record N                                             |  (3 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Where each AFN Size Record is structured as follows:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  AFN                          |  (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  AdrSize      |                  (1 byte)
+-+-+-+-+-+-+-+-+
```

- o Type: AFN-Size sub-sub-TLV type, set to 1 (AFNsz).
- o Length: 3*n where n is the number of AFN Size Records present. If Length is not a multiple of 3, the sub-sub-TLV MUST be ignored.
- o AFN Size Record(s): Zero or more 3-byte records, each giving the size of an address type identified by an AFN,
- o AFN: The AFN whose length is being specified by the AFN Size Record.
- o AdrSize: The length in bytes of addresses specified by the AFN field as an unsigned integer.

An AFN Size sub-sub-TLV for any AFN known to the receiving TRILL switch is compared with the size known to the TRILL switch. If they differ the IA APPsub-TLV is assumed to be corrupt and MUST be ignored.

3.2 Fixed Address sub-sub-TLV

There may be cases where, in a particular Interface Addresses APP-subTLV, the same address would appear in every address set across the APP-subTLV. To avoid wasted space, this sub-sub-TLV can be used to indicate such a fixed address. The address or addresses incorporated into the sets by this sub-sub-TLV are NOT mentioned in the IA APPsub-TLV Template.

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Type=FIXEDADR                 | (2 byte)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Length                        | (2 byte)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| AFN                           | (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Fixed Address                   (variable)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
```

- o Type: Data Label sub-sub-TLV type, set to 2 (FIXEDADR).

- o Length: variable, minimum 2. If Length is 0 or 1 or less, the sub-sub-TLV MUST be ignored.

- o AFN: Address Family Number of the Fixed Address.

- o Fixed Address: The address of the type indicated by the preceding AFN field that is considered to be part of every Address Set in the IA APPsub-TLV.

The Length field implies a size for the Fixed Address. If that size differs from the size of the address type for the given AFN as known by the receiving TRILL switch, the Fixed Address sub-sub-TLV is considered corrupt and MUST be ignored.

3.3 Data Label sub-sub-TLV

This sub-sub-TLV indicates the Data Label within which the interfaces listed in the IA APPsub-TLV are reachable. It is useful if the IA APPsub-TLV occurs outside of the context of a message specifying the Data Label or if it is desired and permitted to override that specification. Multiple occurrences of this sub-sub-TLV indicate that the interfaces are reachable in all of the Data Labels given.

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Type=DATALEN                   | (2 byte)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Length                        | (2 byte)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Data Label                      (variable)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
```

- o Type: Data Label sub-TLV type, set to 3 (LABEL).

- o Length: 2 or 3. If Length is some other value, the sub-sub-TLV MUST be ignored.

- o Data Label: If length is 2, the bottom 12 bits of the Data Label are a VLAN ID and the top 4 bits are reserved (MUST be sent as zero and ignored on receipt). If the length is 3, the three Data Label bytes contain an FGL [RFC7172].

3.4 Topology sub-sub-TLV

The presence of this sub-sub-TLV indicates that the interfaces given in the IA APPsub-TLV are reachable in the topology give. It is useful if the IA APPsub-TLV occurs outside of the context of a message indicating the topology or if it is desired and permitted to override that specification. If it occurs multiple times, then the Address Sets are in all of the topologies given.

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Type=DATALEN                   |  (2 byte)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Length                        |  (2 byte)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| RESV  |        Topology       |  (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

o Type: Topology sub-TLV type, set to 4 (TOPOLOGY).

o Length: 2. If Length is some other values, the sub-sub-TLV MUST be ignored.

RESV: Four reserved bits. MUST be sent as zero and ignored on receipt.

o Topology: The 12-bit topology number [RFC5120].

4. Security Considerations

The integrity of address mapping and reachability information and the correctness of Data Labels (VLANs or FGLs [RFC7172]) are very important. Forged, altered, or incorrect address mapping or Data Labeling can lead to delivery of packets to the incorrect party, violating security policy. However, this document merely describes a data format and does not provide any explicit mechanisms for securing that information, other than a few trivial consistency checks that might detect some corrupted data. Security on the wire, or in storage, for this data is to be providing by the transport or storage used. For example, when transported with ESADI [RFC7357] or RBridge Channel [RFC7178], ESADI security or Channel Tunnel [ChannelTunnel] security mechanisms can be used, respectively.

The address mapping and reachability information, if known to be complete and correct, can be used to detect some cases of forged packet source addresses [RFC7067]. In particular, if native traffic from an end station is received by a TRILL switch that would otherwise accept it but authoritative data indicates the source address should not be reachable from the receiving TRILL switch, that traffic should be discarded. The data format specified in this document may optionally include TRILL switch Port ID number so that this forged address filtering can be optionally applied with port granularity.

See [RFC6325] for general TRILL Security Considerations.

5. IANA Considerations

The following subsections specify IANA actions.

5.1 AFN Number Allocation

IANA has allocated the following AFN values that may be particularly useful for IA APPsub-TLVs:

Hex	Decimal	Description	References
0001	1	IPv4	
0002	2	IPv6	
4005	16389	48-bit MAC	[RFC7042]
4006	16390	64-bit MAC	[RFC7042]
4007	16391	OUI	This document.
4008	16392	MAC/24	This document.
4009	16393	MAC/40	This document.
400A	16394	IPv6/64	This document.
400B	16395	RBridge Port ID	This document.

Other AFNs can be found at http://www.iana.org/assignments/address-family-numbers

The OUI AFN is provided so that MAC addresses can be abbreviated if they have the same upper 24 bits. A MAC/24 is a 24-bit suffix intended to be pre-fixed by an OUI to create a 48-bit MAC address [RFC7042]; in the absence of an OUI, a MAC/24 entry cannot be used. A MAC/40 is a suffix intended to be pre-fixed by an OUI to create a 64-bit MAC address [RFC7042]; in the absence of an OUI, a MAC/40 entry cannot be used.

Typically, an OUI would be provided as a Fixed Address sub-sub-TLV (see Section 3.2).

After Fixed Address sub-sub-TLV processing above, each address set is processed by combining each OUI in the address set with each MAC/24 and each MAC/40 address in the address set. Depending on how many of each of these address types is present, zero or more 48-bit and/or 64-bit MAC addresses may be produced that are considered to be part of the address set. If there are no MAC/24 or MAC/40 addresses present, any OUI’s are ignored. If there are no OUIs, any MAC/24 and/or MAC/40s are ignored. If there are K1 OUIs, K2 MAC/24s, and K3 MAC/40s, K1*K2 48-bit MACs are synthesized and K1*K3 64-bit MACs are synthesized.

IPv6/64 is an 8-byte quantity that is the first 64 bits of an IPv6 address. IPv6/64s are ignored unless, after the processing above in this sub-section, there are one or more 48-bit and/or 64-bit MAC addresses in the address set to provide the lower 64 bits of the IPv6 address. For this purpose, an 48-bit MAC address is expanded to 64 bits as described in [RFC7042]. If there are K4 IPv6/64s present and K5 48- and 64-bit MAC addresses present, K4*K5 128-bit IPv6 addresses are synthesized.

5.2 IA APPsub-TLV Sub-Sub-TLVs SubRegistry

IANA is requested to establish a new subregistry of the TRILL Parameter Registry for sub-sub-TLVs of the Interface Addresses APPsub-TLV with initial contents as shown below.

Name: Interface Addresses APPsub-TLV Sub-Sub-TLVs

Procedure: Expert Review

Note: Types greater than 255 are not usable in some contexts.

Reference: This document

Type	Description	Reference
0	Reserved	
1	AFN Size	This document
2	Fixed Address	This document
3	Data Label	This document
4	Topology	This document
5-254	Available	
255	Reserved	
256-65534	Available	
65535	Reserved	

5.3 IA APPsub-TLV Number

IANA has allocated TBD1 as the Type for the IA APPsub-TLV in the "TRILL APPsub-TLV Types under IS-IS TLV 251 Application Identifier 1" registry from the range under 256. In the registry the Name is "IA" and the Reference is this document.

Acknowledgments

The authors gratefully acknowledge the contributions and review by the following:

Linda Dunbar

The document was prepared in raw nroff. All macros used were defined within the source file.

Appendix A: Examples

Below are example IA APPsub-TLVs.

A.1 Simple Example

Below is an annotated IA APPsub-TLV carrying two simple pairs of EUI-48 MAC addresses and IPv4 addresses from a Push Directory [RFC7042]. No sub-sub-TLVs are included.

```
0x0002(TBD)   Type: Interface Addresses
0x001B        Length: 27 (=0x1B)
0x001B        Address Sets End: 27 (=0x1B)
0x1234        RBridge Nickname from which reachable
0b10000000    Flags: Push Directory data
0xE3          Confidence = 227
35            Template: 35 (0x23) = 31 + 1(MAC48) + 3*1(IPv4)

      Address Set One
0x00005E0053A9   48-bitMAC address
198.51.100.23    IPv4 address

      Address Set Two
0x00005E00536B   48-bit MAC address
203.0.113.201    IPv4 address
```

Size includes 7 for the fixed fields though and including the one byte template, plus 2 times the Address Set size. Each Address Set is 10 bytes, 6 for the 48-bit MAC address plus 4 for the IPv4 address. So total size is 7 + 2*10 = 27.

See Section 2 for more information on Template.

A.2 Complex Example

Below is an annotated IA APPsub-TLV carrying three sets of addresses, each consisting of an EUI-48 MAC address, an IPv4 addresses, an IPv6 address, and an RBridge Port ID, all from a Push Directory [RFC7042]. The IPv6 address for each address set is synthesized from the MAC address given in that set and the IPv6/64 64-bit prefix provided through a Fixed Address sub-sub-TLV. In addition, a sub-sub-TLV is included that provides an FGL which overrides whatever Data Label may be provided by the envelope (for example an ESADI-LSP [RFC7357]) within which this IA APPsub-TLV occurs.

```
      0x0002(TBD)    Type: Interface Addresses
      0x0036         Length: 54 (=0x36)
      0x0021         Address Sets End: 33 (=0x21)
      0x4321         RBridge Nickname from which reachable
      0b10000000     Flags: Push Directory data
      0xD3           Confidence = 211
      72             Template: 72(0x48)=31+1(MAC48)+3*1(IPv4)+36*1(P)

            Address Set One
      0x00005E0053DE   48-bitMAC address
      198.51.100.105   IPv4 address
      0x1DE3           RBridge Port ID

            Address Set Two
      0x00005E0053E3   48-bit MAC address
      203.0.113.89     IPv4 address
      0x1DEE           RBridge Port ID

            Address Set Three
      0x00005E0053D3   48-bit MAC address
      192.0.2.139      IPv4 address
      0x01DE           RBridge Port ID

            sub-sub-TLV One
      0x0003           Type: Data Label
      0x0003           Length: implies FGL
      0xD3E3E3         Fine Grained Label

            sub-sub-TLV Two
      0x0002           Type: Fixed Address
      0x000A           Size: 0x0A = 10
      0x400A           AFN: IPv6/64
      0x20010DB800000000   IPv6 Prefix: 2001:DB8::
```

See Section 2 for more information on Template.

The Fixed Address sub-sub-TLV causes the IPv6/64 value give to be treated as if it occurred as a 4th entry inside each of the three Address Sets. When there is an IPv6/64 entry and a 48-bit MAC entry, the MAC value is expanded by inserting 0xFFFE immediately after the OUI and the resulting 64-bit value is used as the lower 64 bits of the resulting IPv6 address [RFC7042]. As a result, a receiving TRILL switch would treat the three Address Sets shown as if they had an IPv6 address in them as follows:

```
      Address Set One
0x20010DB80000000000005EFFFE0053DE  IPv6 Address

      Address Set Two
0x20010DB80000000000005EFFFE0053E3  IPv6 Address

      Address Set Three
0x20010DB80000000000005EFFFE0053D3  IPv6 Address
```

As an alternative to the compact "well know value" Template encoding used in this example above, the less compact explicit AFN encoding could have been used. In that case, the IA APPsub-TLV would have started as follows:

```
0x0002(TBD)    Type: Interface Addresses
0x003C         Length: 60 (=0x3C)
0x0027         Address Sets End: 39 (=0x27)
0x4321         RBridge Nickname from which reachable
0b10000000     Flags: Push Directory data
0xD3           Confidence = 211
0x3            Template: 3 AFNs
0x4005         AFN: 48-bit MAC
0x0001         AFN: IPv4
0x400B         AFN: RBridge Port ID
```

As a final point, since the 48-bit MAC addresses in these three Address Sets all have the same OUI (the IANA OUI [RFC7042]), it would have been possible to just have a MAC/24 value giving the lower 24 bits of the MAC in each Address Set. The OUI would them be supplied by a second Fixed Address sub-sub-TLV proving the OUI. With N Address Sets, this would have saved 3*N or 9 bytes in this case at the cost of 9 bytes (2 each for the type and length of the sub-sub-TLV, 2 for the OUI AFN number, and 3 for the OUI). So, with just three Address Sets, there would be no net saving; however, with a larger number of Address Sets, there would be a net savings.

Appendix Z: Change History

From -00 to -01

1. Update references for RFC publications.

2. Add this Change History Appendix.

From -01 to -02

1. Fix off-by-one errors in body text and examples for well known Template values.

2. Update for drafts published as RFCs and change in Author Address.

3. Minor editorial improvements.

Normative References

[ISO-10589] - ISO/IEC 10589:2002, Second Edition, "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)", 2002.

[RFC826] - Plummer, D., "An Ethernet Address Resolution Protocol", RFC 826, November 1982.

[RFC903] - Finlayson, R., Mann, T., Mogul, J., and M. Theimer, "A Reverse Address Resolution Protocol", STD 38, RFC 903, June 1984.

[RFC2119] - Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997

[RFC4861] - Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.

[RFC5120] - Przygienda, T., Shen, N., and N. Sheth, "M-ISIS: Multi Topology (MT) Routing in Intermediate System to Intermediate Systems (IS-ISs)", RFC 5120, February 2008.

[RFC5226] - Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

[RFC5305] - Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, October 2008.

[RFC6325] - Perlman, R., Eastlake 3rd, D., Dutt, D., Gai, S., and A. Ghanwani, "Routing Bridges (RBridges): Base Protocol Specification", RFC 6325, July 2011.

[RFC6823] - Ginsberg, L., Previdi, S., and M. Shand, "Advertising Generic Information in IS-IS", RFC 6823, December 2012.

[RFC7042] - Eastlake 3rd, D. and J. Abley, "IANA Considerations and IETF Protocol and Documentation Usage for IEEE 802 Parameters", BCP 141, RFC 7042, October 2013.

[RFC7172] - Eastlake 3rd, D., Zhang, M., Agarwal, P., Perlman, R., and D. Dutt, "Transparent Interconnection of Lots of Links (TRILL): Fine-Grained Labeling", RFC 7172, May 2014.

[RFC7356] - Ginsberg, L., Previdi, S., and Y. Yang, "IS-IS Flooding Scope Link State PDUs (LSPs)", RFC 7356, September 2014, <http://www.rfc-editor.org/info/rfc7356>.

[RFC7357] - Zhai, H., Hu, F., Perlman, R., Eastlake 3rd, D., and O. Stokes, "Transparent Interconnection of Lots of Links (TRILL): End Station Address Distribution Information (ESADI) Protocol", RFC 7357, September 2014, <http://www.rfc-editor.org/info/rfc7357>.

Informational References

[ARP reduction] - Shah, et. al., "ARP Broadcast Reduction for Large Data Centers", draft-shah-armd-arp-reduction, work in progress.

[ChannelTunnel] - D. Eastlake, Y. Li, "TRILL: RBridge Channel Tunnel Protocol", draft-eastlake-trill-channel-tunnel, work in progress.

[DirectoryScheme] - Dunbar, L., D. Eastlake, R. Perlman, I. Gashinsky, Y. Li, "TRILL": Directory Assistance Mechanisms", draft-dunbar-trill-scheme-for-directory-assist, work in progress.

[RFC5494] - Arkko, J. and C. Pignataro, "IANA Allocation Guidelines for the Address Resolution Protocol (ARP)", RFC 5494, April 2009.

[RFC7067] - Dunbar, L., Eastlake 3rd, D., Perlman, R., and I. Gashinsky, "Directory Assistance Problem and High-Level Design Proposal", RFC 7067, November 2013.

[RFC7178] - Eastlake 3rd, D., Manral, V., Li, Y., Aldrin, S., and D. Ward, "Transparent Interconnection of Lots of Links (TRILL): RBridge Channel Support", RFC 7178, May 2014.

Authors' Addresses

Donald Eastlake
Huawei Technologies
155 Beaver Street
Milford, MA 01757 USA

Phone: +1-508-333-2270
Email: d3e3e3@gmail.com

Yizhou Li
Huawei Technologies
101 Software Avenue,
Nanjing 210012 China

Phone: +86-25-56622310
Email: liyizhou@huawei.com

Radia Perlman
EMC
2010 256th Avenue NE, #200
Bellevue, WA 98007 USA

Email: Radia@alum.mit.edu

Copyright, Disclaimer, and Additional IPR Provisions

Network Working Group M. Wasserman
Internet-Draft Painless Security
Intended status: Standards Track D. Eastlake
Expires: August 6, 2015 Huawei Technologies
D. Zhang
Alibaba
February 2, 2015

Transparent Interconnection of Lots of Links (TRILL) over IP
draft-ietf-trill-over-ip-02.txt

Abstract

The Transparent Interconnection of Lots of Links (TRILL) protocol is implemented by devices called TRILL Switches or RBridges (Routing Bridges). TRILL supports both point-to-point and multi-access links and is designed so that a variety of link protocols can be used between TRILL switch ports. This document standardizes methods for encapsulating TRILL in IP (v4 or v6) so as to use IP as a TRILL link protocol in a unified TRILL campus.

Table of Contents

1. Requirements Terminology

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 RFC 2119 [RFC2119].

2. Introduction

TRILL switches (RBridges) are devices that implement the IETF TRILL protocol [RFC6325] [RFC7176] [RFC7177].

RBridges provide transparent forwarding of frames within an arbitrary network topology, using least cost paths for unicast traffic. They support not only VLANs and Fine Grained Labels [RFC7172] but also multipathing of unicast and multi-destination traffic. They use IS-IS link state routing and encapsulation with a hop count.

Ports on different RBridges can communicate with each other over various link types, such as Ethernet [RFC6325], pseudowires [RFC7173], or PPP [RFC6361].

This document defines a method for RBridges to communicate over IP (v4 or v6). TRILL over IP will allow Internet-connected RBridges to form a single TRILL campus, or multiple TRILL over IP networks within a campus to be connected as a single TRILL campus via a TRILL over IP backbone.

TRILL over IP connects RBridge ports using IPv4 or IPv6 as a transport in such a way that the ports appear to TRILL to be connected by a single multi-access link. Therefore, if more than two RBridge ports are connected via a single TRILL over IP link, any pair of them can communicate.

To support the scenarios where RBridges are connected via IP paths (such as over the public Internet) that are not under the same administrative control as the TRILL campus and/or not physically secure, this document specifies the use of IPsec [RFC4301] Encapsulating Security Protocol [RFC4303] to secure all or part of such paths.

3. Use Cases for TRILL over IP

This section introduces two application scenarios (a remote office scenario and an IP backbone scenario) which cover typical situations where network administrators may choose to use TRILL over an IP network to connect TRILL switches.

3.1. Remote Office Scenario

In the Remote Office Scenario, a remote TRILL network is connected to a TRILL campus across a multihop IP network, such as the public Internet. The TRILL network in the remote office becomes a logical part of TRILL campus, and nodes in the remote office can be attached to the same VLANs or Fine Grained Labels[RFC7172] as local campus nodes. In many cases, a remote office may be attached to the TRILL campus by a single pair of RBridges, one on the campus end, and the other in the remote office. In this use case, the TRILL over IP link will often cross logical and physical IP networks that do not support TRILL, and are not under the same administrative control as the TRILL campus.

3.2. IP Backbone Scenario

In the IP Backbone Scenario, TRILL over IP is used to connect a number of TRILL networks to form a single TRILL campus. For example, a TRILL over IP backbone could be used to connect multiple TRILL networks on different floors of a large building, or to connect TRILL networks in separate buildings of a multi-building site. In this use case, there may often be several TRILL switches on a single TRILL over IP link, and the IP link(s) used by TRILL over IP are typically under the same administrative control as the rest of the TRILL campus.

3.3. Important Properties of the Scenarios

There are a number of differences between the above two application scenarios, some of which drive features of this specification. These differences are especially pertinent to the security requirements of the solution, how multicast data frames are handled, and how the TRILL switch ports discover each other.

3.3.1. Security Requirements

In the IP Backbone Scenario, TRILL over IP is used between a number of RBridge ports, on a network link that is in the same administrative control as the remainder of the TRILL campus. While it is desirable in this scenario to prevent the association of rogue RBridges, this can be accomplished using existing IS-IS security mechanisms. There may be no need to protect the data traffic, beyond any protections that are already in place on the local network.

In the Remote Office Scenario, TRILL over IP may run over a network that is not under the same administrative control as the TRILL network. Nodes on the network may think that they are sending traffic locally, while that traffic is actually being sent, in an IP tunnel, over the public Internet. It is necessary in this scenario to protect the integrity and confidentiality of user traffic, as well as ensuring that no unauthorized RBridges can gain access to the RBridge campus. The issues of protecting integrity and confidentiality of user traffic are addressed by using IPsec for both TRILL IS-IS and TRILL Data packets between RBridges in this scenario.

3.3.2. Multicast Handling

In the IP Backbone scenario, native multicast may be supported on the TRILL over IP link. If so, it can be used to send TRILL IS-IS and multicast data packets, as discussed later in this document. Alternatively, multi-destination packets can be transmitted serially by unicast.

In the Remote Office Scenario there will often be only one pair of RBridges connecting a given site and, even when multiple RBridges are used to connect a Remote Office to the TRILL campus, the intervening network may not provide reliable (or any) multicast connectivity. Issues such as complex key management also make it difficult to provide strong data integrity and confidentiality protections for multicast traffic. For all of these reasons, the connections between local and remote RBridges will commonly be treated like point-to-point links, and all TRILL IS-IS control messages and multicast data packets that are transmitted between the Remote Office and the TRILL campus will be serially transmitted by unicast, as discussed later in this document.

3.3.3. RBridge Neighbor Discovery

In the IP Backbone Scenario, RBridges that use TRILL over IP will use the normal TRILL IS-IS Hello mechanisms to discover the existence of other RBridges on the link [RFC7177], and to establish authenticated communication with those RBridges.

In the Remote Office Scenario, an IPsec session will need to be established before TRILL IS-IS traffic can be exchanged, as discussed below. In this case, one end will need to be configured to establish a IPSEC session with the other. This will typically be accomplished by configuring the RBridge or a border device at a Remote Office to initiate an IPsec session and subsequent TRILL exchanges with a TRILL over IP-enabled RBridge attached to the TRILL campus.

4. TRILL Packet Formats

To support the TRILL base protocol standard [RFC6325], two types of packets will be transmitted between RBridges: TRILL Data packets and TRILL IS-IS packets.

The on-the-wire form of a TRILL Data packet in transit between two neighboring RBridges is as shown below:

```
+--------------+----------+---------------+-----------+
| TRILL Data   |  TRILL   |  Native Frame  |   Link    |
| Link Header  |  Header  |    Payload     |  Trailer  |
+--------------+----------+---------------+-----------+
```

Where the Encapsulated Native Frame Payload is similar to Ethernet frame format with a VLAN tag or Fine Grained Label [RFC7172] but with no trailing Frame Check Sequence (FCS).

TRILL IS-IS packets are formatted on-the-wire as follows:

```
+--------------+--------------+-----------+
| TRILL IS-IS  |  TRILL IS-IS  |   Link    |
| Link Header  |    Payload    |  Trailer  |
+--------------+--------------+-----------+
```

The Link Header and Link Trailer in these formats depend on the specific link technology. The Link Header contains one or more fields that distinguish TRILL Data from TRILL IS-IS. For example, over Ethernet, the TRILL Data Link Header ends with the TRILL Ethertype while the TRILL IS-IS Link Header ends with the L2-IS-IS Ethertype; on the other hand, over PPP, there are no Ethertypes but PPP protocol code points are included that distinguish TRILL Data from TRILL IS-IS.

In TRILL over IP, we will use UDP/IP (v4 or v6) as the link header, and the TRILL packet type will be determined based on the UDP destination port number. In TRILL over IP, no Link Trailer is specified, although one may be added when the resulting IP packets are encapsulated for transmission on a network (e.g. Ethernet).

5. Link Protocol Specifics

TRILL Data packets can be unicast to a specific RBridge or multicast to all RBridges on the link. TRILL IS-IS packets are always multicast to all other RBridge on the link (except for MTU PDUs, which may be unicast [RFC7177]). On Ethernet links, the Ethernet multicast address All-RBridges is used for TRILL Data and All-IS-IS-RBridges for TRILL IS-IS.

To properly handle TRILL base protocol packets on a TRILL over IP link, either native multicast mode must be used on that link, or multicast must be simulated using serial unicast, as discussed below.

In TRILL Hello PDUs used on TRILL IP links, the IP addresses of the connected IP ports are their real SNPA (SubNetwork Point of Attachment [IS-IS]) addresses and, for IPv6, the 16-byte IPv6 address is used; however, for easy of code re-use designed for common 48-bit SNPAs, for TRILL over IPv4, a 48-bit synthetic SNPA that looks like a unicast MAC address is constructed for use in the SNPA field of TRILL Neighbor TLVs [RFC7176][RFC7177] on the link. This synthetic SNPA is as follows:

```
                       1 1 1 1 1 1
   0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |  0xFE          |  0x00         |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |  IPv4 upper half               |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |  IPv4 lower half               |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

This synthetic SNPA/MAC address has the local (0x02) bit on in the first byte and so cannot conflict with any globally unique 48-bit Ethernet MAC. However, at the IP level, where TRILL operates on an IP link, there are only IP stations, not MAC stations, so conflict on the link with a real MAC address would be impossible in any case.

6. RBridge IP Port Configuration

This section specifies the configuration information needed at a TRILL over IP port beyond that needed for a general RBridge port.

6.1. Per IP Port Configuration

Each RBridge port used for a TRILL over IP link should have at least one IP (v4 or v6) address. If no IP address is associated with the port, perhaps as a transient condition during re-configuration, the port is disabled. Implementations MAY allow a single port to operate as multiple IPv4 and/or IPv6 logical ports. Each IP address constitutes a different logical port and the RBridge with those ports MUST associate a different Port ID with each logical port.

By default an RBridge IP port discards output packets that fail the possible recursive ingress test (see Section 10.1) unless configured to disable that test.

6.2. Additional per IP Address Cofiguration

The configuration information specified below is per IP address at a TRILL over IP port.

Each IP address at a TRILL over IP port uses native IP multicast by default but may be configured whether to use serial unicast (Section 6.2.2) or native multicast (Section 6.2.1). Each IP address at a TRILL over IP is configured whether or not to use IPsec (Section 6.2.3).

6.2.1. Native Multicast Configuration

If a TRILL IP port address is using native IP multicast for multi-destination TRILL packets (IS-IS and data), by default transmissions from that IP address use the appropriate IP multicast address (IPv4 or IPv6) specified in Section 13.2. The RBridge IP port may be configured to use a different IP multicast address or multi-destination packets.

6.2.2. Serial Unicast Configuration

If a TRILL over IP port address has been configured to use serial unicast for multi-destination packets (IS-IS and data), it should have associated with it a non-empty list of unicast IP destination addresses. Multi-destination TRILL packets are serially unicast to the addresses in this list. Such a TRILL over IP port will only be able to form adjacencies [RFC7177] with the RBridges at the addresses in this list as those are the only RBridges to which it will send TRILL Hellos.

If the list is empty, there is no way to transmit a multi-destination TRILL over IP packet such as a TRILL Hello. Thus it is impossible to achieve adjacency [RFC7177] or if adjacency had been achieved (perhaps the list was non-empty and has just been configured to be empty), no way to maintain such adjacency. Thus, in the empty list case, TRILL Data multi-destination packets cannot be sent and TRILL Data unicast packets will not start flowing or, if they are already flowing, will soon cease.

6.2.3. Security Configuration

... tbd ...

7. TRILL over IP Format

The general format of a TRILL over IP packet without security is shown below.

```
+----------+--------+----------------------+
| IP       | UDP    |  TRILL               |
| Header   | Header |  Payload             |
+----------+--------+----------------------+
```

Where the UDP Header is as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Source Port = Entropy      |      Destination Port         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           UDP Length          |        UDP Checksum           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  TRILL Payload ...
```

Source Port - see Section 10.2

Destination Port - indicates TRILL Data or IS-IS, see Section 14

UDP Length - as specified in [RFC0768]

UDP Checksum - as specified in [RFC0768]

The TRILL Payload starts with the TRILL Header (not including the TRILL Ethertype) for TRILL Data packets and starts with the 0x83 Intradomain Routeing Protocol Discriminator byte (thus not including the L2-IS-IS Ethertype) for TRILL IS-IS packets.

TRILL over IP link security uses IPsec Encapsulating Security Protocol (ESP) in tunnel mode. The resulting packet format is as follows for IPv4 and IPv6:

```
          ------------------------------------------------------------
   IPv4   | new IP hdr  |     | orig IP hdr   |   |TRILL| ESP   | ESP|
          |(any options)| ESP | (any options) |UDP|Data |Trailer| ICV|
          ------------------------------------------------------------
                              |<--------- encryption ---------->|
                        |<------------- integrity ------------->|

          -------------------------------------------------------------
   IPv6   | new  |new ext |   | orig |orig ext |   |TRILL| ESP   | ESP|
          |IP hdr| hdrs   |ESP|IP hdr| hdrs    |UDP|Data |Trailer| ICV|
          -------------------------------------------------------------.
                              |<--------- encryption ----------->|
                          |<------------ integrity ------------->|
```

This architecture permits the ESP tunnel termination to be separated from the TRILL over IP RBridge port and, for example, placed at a physical or administrative security boundary. If two or more RBridge TRILL over IP ports are communicate securely using IPsec, there are three possibilities:

(a) For all ports involved, the IPsec implementation is integrated with the RBridge port. In this case it is straightforward to use the default and negotiations specified herein for keying and algorithms.

(b) Some of the IPsec implementations are integrated with an RBridge port and some are not. For example, on a point-to-point TRILL over IP link, IPsec could be integrated with the RBridge port at one end but implemented in a separate appliances that could be separated by IP routers from the TRILL over IP RBridge port at the other end. In this case mechanisms beyond the scope of this document may be required to communicate default or negotiated keying or algorithms between such separate appliances and the RBridge port for which they are providing TRILL over IP security services.

(c) For all ports involved, the IPsec implementation is in a separate appliance. In this case, if adequate security is provided, the appliances MAY negotiation IPsec keying and algorithms as they see fit. Alternatively, the specifications of this document for keying and algorithms are used and mechanisms beyond the scope of this document may be required to communicate default or negotiated keying or algorithms between such separate appliances and the RBridge port for which they are providing TRILL over IP security services

8. Handling Multicast

By default, both TRILL IS-IS packets and multi-destination TRILL Data packets are sent to an All-RBridges IPv4 or IPv6 multicast Address as appropriate (see Section 13.2); however, a TRILL over IP port may be configured (see Section 6) to use serial unicast with a list of one or more unicast IP addresses of other TRILL over IP ports to which multi-destination packets are sent. Such configuration is necessary if the TRILL over IP port is connected to an IP network that does not support IP multicast. In both cases, unicast TRILL data packets would be sent by unicast IP.

When a TRILL over IP port is using IP multicast, it MUST periodically transmit appropriate IGMP (IPv4 [RFC3376]) or MLD (IPv6 [RFC2710]) packets so that the TRILL multicast IP traffic will be sent to it.

Although TRILL fully supports broadcast links with more than 2 RBridges connected to the link, even where native IP multicast is available, there may be good reasons for configuring TRILL over IP ports to use serial unicast. In some networks, unicast is more reliable than multicast. If multiple unicast connections between parts of a TRILL campus are configured, TRILL will in any case spread traffic across them, treating them as parallel links, and appropriately fail over traffic if a link ceases to operate or incorporate a new link that comes up.

9. Use of IPsec

All RBridges that support TRILL over IP MUST implement IPsec and support the use of IPsec Encapsulating Security Protocol (ESP) to secure both TRILL IS-IS and TRILL data packets. When IPsec is used to secure a TRILL over IP link and no IS-IS security is enabled, the IPsec session MUST be fully established before any TRILL IS-IS or data packets are exchanged. When there is IS-IS security [RFC5310] provided, people may select to use IS-IS security to protect TRILL IS-IS packets. However, in this case, the IPsec session still MUST be fully established before any data packets transmission since IS-IS security does not provide any protection to data packets.

... TBD ...

9.1. Default Pre-Shared Keys

The default pre-shared keyes for IPsec usage are derived as follows:

HMAC-SHA256 ("TRILL IP"| IS-IS-shared key)

In the above "|" indicates concatenation, HMAC-SHA256 is as described in [FIPS180] [RFC6234] and "TRILL IP" is the eight byte US ASCII [RFC0020] string indicated. IS-IS-shared key is a link (or wider scope) IS-IS key usable for IS-IS security of link local IS-IS local PDUs such as Hello, CSNP, and PSNP. With [RFC5310]there could be multiple keys identified with 16-bit key IDs. In this case, the Key ID of IS-IS-shared key is also used to identify the derived key.

10. Transport Considerations

This section discusses a variety of transport considerations.

10.1. Recursive Ingress

TRILL is designed to transport end station traffic to and from end stations over IEEE 802.3 and IP is frequently transported over IEEE 802.3 or similar protocols. Thus, an end station native data frame EF might get TRILL ingressed to TRILL(EF) which was then sent on a TRILL over IP over an 802.3 link resulting in an 802.3 frame of the form 802.3(IP(TRILL(EF))). There is a risk of such a packet being re-ingressed by the same TRILL campus, due to physical or logical misconfiguration, looping round, being further re-ingressed, etc. The packet might get discarded if it got too large but if fragmentation is enabled, it would just keep getting split into fragments that would continue to loop and grow and re-fragment until the path was saturated with junk and packets were being discarded due to queue overflow. The TRILL Header TTL would provide no protection because each TRILL ingress adds a new Header and TTL.

To protect against this scenario, a TRILL over IP port MUST by, default, test whether a TRILL packet it is about to send is, in fact a TRILL ingress of a TRILL over IP over 802.3 or the like packets. That is, is it of the form TRILL(802.3(IP(TRILL(...)))? If so, the default action of the TRILL over IP output port is to discard the packet rather than transmit it. However, there are cases where some level of nested ingress is desired so it MUST be possible to configure the port to allow such packets.

10.2. Fat Flows

For the purpose of load balancing, it is worthwhile to consider how to transport the TRILL packets over the Equal Cost Multiple Paths (ECMPs) existing in the IP path.

The ECMP election for the IP traffics could be based, at least for IPv4, on the quintuple of the outer IP header { Source IP, Destination IP, Source Port, Destination Port, and IP protocol }. Such tuples, however, could be exactly the same for all TRILL Data packets between two RBridge ports, even if there is a huge amount of data being sent between a variety of ingress and egress RBridges. Therefore, in order to better support ECMP, a RBridge SHOULD set the Source Port as an entropy field for ECMP decisions. (This idea is also introduced in [I-D.yong-tsvwg-gre-in-udp-encap].For example, for

TRILL Data this entropy field could be based on the Inner.MacDA, Inner.MacSA, and Inner.VLAN or Inner.FGL.

10.3. Congestion Considerations

Section 3.1.3 of [RFC5405] discussed the congestion implications of UDP tunnels. As discussed in [RFC5405], because other flows can share the path with one or more UDP tunnels, congestion control [RFC2914] needs to be considered.

One motivation for encapsulating TRILL in UDP is to improve the use of multipath (such as ECMP) in cases where traffic is to traverse routers which are able to hash on UDP Port and IP address. In many cases this may reduce the occurrence of congestion and improve usage of available network capacity. However, it is also necessary to ensure that the network, including applications that use the network, responds appropriately in more difficult cases, such as when link or equipment failures have reduced the available capacity.

The impact of congestion must be considered both in terms of the effect on the rest of the network of a UDP tunnel that is consuming excessive capacity, and in terms of the effect on the flows using the UDP tunnels. The potential impact of congestion from a UDP tunnel depends upon what sort of traffic is carried over the tunnel, as well as the path of the tunnel.

TRILL is used to carry a wide range of traffic. In many cases TRILL is used to carry IP traffic. IP traffic is generally assumed to be congestion controlled, and thus a tunnel carrying general IP traffic (as might be expected to be carried across the Internet) generally does not need additional congestion control mechanisms. As specified in [RFC5405]:

"IP-based traffic is generally assumed to be congestion- controlled, i.e., it is assumed that the transport protocols generating IP-based traffic at the sender already employ mechanisms that are sufficient to address congestion on the path. Consequently, a tunnel carrying IP-based traffic should already interact appropriately with other traffic sharing the path, and specific congestion control mechanisms for the tunnel are not necessary".

For this reason, where TRILL is tunneled through UDP and used to carry IP traffic that is known to be congestion controlled, the UDP tunnels MAY be used across any combination of a single or cooperating service providers or across the general Internet.

However, TRILL is also used to carry traffic that is not necessarily congestion controlled. For example, TRILL may be used to carry traffic where specific bandwidth guarantees are provided.

In such cases congestion may be avoided by careful provisioning of the network and/or by rate limiting of user data traffic. Where TRILL is carried, directly or indirectly, over UDP over IP, the identity of each individual TRILL flow is in general lost.

For this reason, where the TRILL traffic is not congestion controlled, TRILL over UDP/IP MUST only be used within a single service provider that utilizes careful provisioning (e.g., rate limiting at the entries of the network while over-provisioning network capacity) to ensure against congestion, or within a limited number of service providers who closely cooperate in order to jointly provide this same careful provisioning. As such, TRILL over USP/IP MUST NOT be used over the general Internet, or over non-cooperating service providers, to carry traffic that is not congestion-controlled.

Measures SHOULD be taken to prevent non-congestion-controlled TRILL over UDP/IP traffic from "escaping" to the general Internet, for example the following:

a. Physical or logical isolation of the TRILL over IP links from the general Internet.

b. Deployment of packet filters that block the UDP ports assigned for TRILL-over-UDP.

c. Imposition of restrictions on TRILL over UDP/IP traffic by software tools used to set up TRILL over UDP paths between specific end systems (as might be used within a single data center).

d. Use of a "Managed Circuit Breaker" for the TRILL traffic as described in [I-D.ietf-tsvwg-circuit-breaker].

10.4. MTU Considerations

In TRILL each RBridge advertises in its LSP number zero the largest LSP frame it can accept (but not less than 1,470 bytes) on any of its interfaces (at least those interfaces with adjacencies to other RBridges in the campus) through the originatingLSPBufferSize TLV [RFC6325] [RFC7177]. The campus minimum MTU, denoted Sz, is then established by taking the minimum of this advertised MTU for all RBridges in the campus. Links that do not meet the Sz MTU are not included in the routing topology. This protects the operation of IS-IS from links that would be unable to accommodate some LSPs.

A method of determining originatingLSPBufferSize for an RBridge with one or more TRILL over IP portsis described in [RFC7180]. However, if an IP link either can accommodate jumbo frames or is a link on which IP fragmentation is enabled and acceptable, then it is unlikely that the IP link will be a constraint on the originatingLSPBufferSize of an RBridge using the link. On the other hand, if the IP link can only handle smaller frames and fragmentation is to be avoided when possible, a TRILL over IP port might constrain the RBridge’s originatingLSPBufferSize. Because TRILL sets the minimum values of Sz at 1,470 bytes, there may be links that meet the minimum MTU for the IP protocol (1,280 bytes for IPv6, theoretically 68 bytes for IPv4) on which it would be necessary to enable fragmentation for TRILL use.

The optional use of TRILL IS-IS MTU PDUs, as specified in [RFC6325] and [RFC7177] can provide added assurance of the actual MTU of a link.

11. Middlebox Considerations

... TBD ...

12. Security Considerations

TRILL over IP is subject to all of the security considerations for the base TRILL protocol [RFC6325]. In addition, there are specific security requirements for different TRILL deployment scenarios, as discussed in the "Use Cases for TRILL over IP" section above.

This document specifies that all RBridges that support TRILL over IP MUST implement IPsec, and makes it clear that it is both wise and good to use IPsec in all cases where a TRILL over IP link will traverse a network that is not under the same administrative control as the rest of the TRILL campus or is not physically secure. IPsec is necessary, in these cases to protect the privacy and integrity of data traffic.

TRILL over IP is completely compatible with the use of IS-IS Security [RFC5310], which can be used to authenticate RBridges before allowing them to join a TRILL campus. This is sufficient to protect against rogue RBridges, but is not sufficient to protect data packets that may be sent in IP outside of the local network, or even across the public Internet. To protect the privacy and integrity of that traffic, use IPsec.

In cases were IPsec is used, the use of IS-IS security may not be necessary, but there is nothing about this specification that would prevent using both IPsec and IS-IS security together. In cases where both types of security are enabled, by default, a key derived from the IS-IS key will be used for IPsec.

13. IANA Considerations

IANA considerations are given below.

13.1. Port Assignments

IANA has allocated the following destination UDP Ports for the TRILL IS-IS and Data channels:

UDP Port	Protocol
(TBD)	TRILL IS-IS Channel
(TBD)	TRILL Data Channel

13.2. Multicast Address Assignments

IANA has allocated one IPv4 and one IPv6 multicast address, as shown below, which correspond to the All-RBridges and All-IS-IS-RBridges multicast MAC addresses that the IEEE Registration Authority has assigned for TRILL. Because the low level hardware MAC address dispatch considerations for TRILL over Ethernet do not apply to TRILL over IP, one IP multicast address for each version of IP is sufficient.

[Values recommended to IANA:]

Name	IPv4	IPv6
All-RBridges	233.252.14.0	FF0X:0:0:0:0:0:0:205

Note: when these IPv4 and IPv6 multicast addresses are used and the resulting IP frame is sent over Ethernet, the usual IP derived MAC address is used.

[Need to discuss scopes for IPv6 multicast (the "X" in the addresses) somewhere. Default to "site" scope but MUST be configurable?]

14. Acknowledgements

This document was written using the xml2rfc tool described in RFC 2629 [RFC2629].

The following people have provided useful feedback on the contents of this document: Sam Hartman, Adrian Farrel.

Some material in Section 10.2 is derived from draft-ietf-mpls-in-udp by Xiaohu Xu, Nischal Sheth, Lucy Yong, Carlos Pignataro, and Yongbing Fan.

15. References

15.1. Normative References

[FIPS180] ""Secure Hash Standard (SHS)", United States of American, National Institute of Science and Technology, Federal Information Processing Standard (FIPS) 180-4", March 2012.

[IS-IS] ""Intermediate system to Intermediate system routeing information exchange protocol for use in conjunction with the Protocol for providing the Connectionless-mode Network Service (ISO 8473)", ISO/IEC 10589:2002.", 2002.

[RFC0020] Cerf, V., "ASCII format for network interchange", RFC 20, October 1969.

[RFC0768] Postel, J., "User Datagram Protocol", STD 6, RFC 768, August 1980.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2710] Deering, S., Fenner, W., and B. Haberman, "Multicast Listener Discovery (MLD) for IPv6", RFC 2710, October 1999.

[RFC2914] Floyd, S., "Congestion Control Principles", BCP 41, RFC 2914, September 2000.

[RFC3376] Cain, B., Deering, S., Kouvelas, I., Fenner, B., and A. Thyagarajan, "Internet Group Management Protocol, Version 3", RFC 3376, October 2002.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

[RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, December 2005.

[RFC5310] Bhatia, M., Manral, V., Li, T., Atkinson, R., White, R., and M. Fanto, "IS-IS Generic Cryptographic Authentication", RFC 5310, February 2009.

[RFC5405] Eggert, L. and G. Fairhurst, "Unicast UDP Usage Guidelines for Application Designers", BCP 145, RFC 5405, November 2008.

[RFC6325] Perlman, R., Eastlake, D., Dutt, D., Gai, S., and A. Ghanwani, "Routing Bridges (RBridges): Base Protocol Specification", RFC 6325, July 2011.

[RFC7176] Eastlake, D., Senevirathne, T., Ghanwani, A., Dutt, D., and A. Banerjee, "Transparent Interconnection of Lots of Links (TRILL) Use of IS-IS", RFC 7176, May 2014.

[RFC7177] Eastlake, D., Perlman, R., Ghanwani, A., Yang, H., and V. Manral, "Transparent Interconnection of Lots of Links (TRILL): Adjacency", RFC 7177, May 2014.

[RFC7180] Eastlake, D., Zhang, M., Ghanwani, A., Manral, V., and A. Banerjee, "Transparent Interconnection of Lots of Links (TRILL): Clarifications, Corrections, and Updates", RFC 7180, May 2014.

15.2. Informative References

[I-D.ietf-tsvwg-circuit-breaker]
Fairhurst, G., "Network Transport Circuit Breakers", draft-ietf-tsvwg-circuit-breaker-00 (work in progress), September 2014.

[I-D.yong-tsvwg-gre-in-udp-encap]
Crabbe, E., Yong, L., and X. Xu, "Generic UDP Encapsulation for IP Tunneling", draft-yong-tsvwg-gre-in-udp-encap-02 (work in progress), October 2013.

[RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, June 1999.

[RFC6234] Eastlake, D. and T. Hansen, "US Secure Hash Algorithms (SHA and SHA-based HMAC and HKDF)", RFC 6234, May 2011.

[RFC6361] Carlson, J. and D. Eastlake, "PPP Transparent Interconnection of Lots of Links (TRILL) Protocol Control Protocol", RFC 6361, August 2011.

[RFC7172] Eastlake, D., Zhang, M., Agarwal, P., Perlman, R., and D. Dutt, "Transparent Interconnection of Lots of Links (TRILL): Fine-Grained Labeling", RFC 7172, May 2014.

[RFC7173] Yong, L., Eastlake, D., Aldrin, S., and J. Hudson, "Transparent Interconnection of Lots of Links (TRILL) Transport Using Pseudowires", RFC 7173, May 2014.

Authors' Addresses

Margaret Wasserman
Painless Security
356 Abbott Street
North Andover, MA 01845
USA

Phone: +1 781 405-7464
Email: mrw@painless-security.com
URI: http://www.painless-security.com

Donald Eastlake
Huawei Technologies
155 Beaver Street
Milford, MA 01757
USA

Phone: +1 508 333-2270
Email: d3e3e3@gmail.com

Dacheng Zhang
Alibaba
Beijing, Chao yang District
P.R. China

Email: dacheng.zdc@alibaba-inc.com

TRILL Working Group
Internet-Draft
Intended Status: Standards Track
Expires: September 10, 2015

H. Zhai
JIT
T. Senevirathne
Cisco Systems
R. Perlman
EMC
M. Zhang
Y. Li
Huawei Technologies

March 9, 2015

TRILL: Pseudo-Nickname for Active-Active Access
draft-ietf-trill-pseudonode-nickname-04

Abstract

The IETF TRILL (TRansparent Interconnection of Lots of Links) protocol provides support for flow level multi-pathing for both unicast and multi-destination traffic in networks with arbitrary topology. Active-active access at the TRILL edge is the extension of these characteristics to end stations that are multiply connected to a TRILL campus as discussed in RFC 7379. In this document, the edge RBridge (TRILL switch) group providing active-active access to such an end station are represented as a Virtual RBridge. Based on the concept of Virtual RBridge along with its pseudo-nickname, this document specifies a method for TRILL active-active access by such end stations.

Status of This Memo

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

Copyright and License Notice

Table of Contents

1. Introduction

The IETF TRILL protocol [RFC6325] provides optimal pair-wise data frame forwarding without configuration, safe forwarding even during periods of temporary loops, and support for multi-pathing of both unicast and multicast traffic. TRILL accomplishes this by using IS-IS [IS-IS] [RFC7176] link state routing and encapsulating traffic using a header that includes a hop count. Devices that implement TRILL are called RBridges or TRILL switches.

In the base TRILL protocol, an end node can be attached to the TRILL campus via a point-to-point link or a shared link such as a bridged LAN (Local Area Network). Although there might be more than one edge RBridge on a shared link, to avoid potential forwarding loops, one and only one of the edge RBridges is permitted to provide forwarding service for end station traffic in each VLAN (Virtual LAN). That RBridge is referred to as the Appointed Forwarder (AF) for that VLAN on the link [RFC6325] [RFC6439]. However, in some practical deployments, to increase the access bandwidth and reliability, an end station might be multiply connected to several edge RBridges and all of the uplinks are handled via a Local Active-Active Link Protocol (LAALP [RFC7379]) such as Multi-Chassis Link Aggregation (MC-LAG) or Distributed Resilient Network Interconnect (DRNI [802.1AX]). In this case, it’s required that traffic can be ingressed/egressed into/from the TRILL campus by any of the RBridges for each given VLAN. These RBridges constitutes an Active-Active Edge (AAE) RBridge group.

With an LAALP, traffic with the same VLAN and source MAC address but belonging to different flows will frequently be sent to different member RBridges of the AAE group and then ingressed into TRILL campus. When an egress RBridge receives such TRILL data packets ingressed by different RBridges, it learns different VLAN and MAC address to nickname correspondences continuously when decapsulating the packets if it has data plane address learning enabled. This issue is known as the "MAC flip-flopping" issue, which makes most TRILL switches behave badly and causes the returning traffic to reach the destination via different paths resulting in persistent re-ordering of the frames. In addition to this issue, other issues such as duplicate egressing and loop back of multi-destination frames may also disturb an end station multiply connected to the member RBridges of an AAE group [RFC7379].

This document addresses the AAE issues of TRILL by specifying how members of an edge RBridge group can be represented by a Virtual RBridge (RBv) and assigned a pseudo-nickname. A member RBridge of such a group uses a pseudo-nickname, instead of its own nickname, as the ingress RBridge nickname when ingressing frames received on attached LAALP links. Other methods are possible; for example the specification in this document and the specification in [MultiAttach] could be simultaneously deployed for different AAE groups in the same campus.

The main body of this document is organized as follows: Section 2 gives an overview of the TRILL active-active access issues and the reason that a virtual RBridge (RBv) is used to resolve the issues. Section 3 gives the concept of a virtual RBridge (RBv) and its pseudo-nickname. Section 4 describes how edge RBridges can support an RBv automatically and get a pseudo-nickname for the RBv. Section 5 discusses how to protect multi-destination traffic against disruption due to Reverse Forwarding Path (RPF) check failure, duplication, forwarding loops, etc. Section 6 covers the special processing of native frames and TRILL data packets at member RBridges of an RBv (also referred to as an Active-Active Edge (AAE) RBridge group). Section 7 describes the MAC information synchronization among the member RBridges of an RBv. Section 8 discusses protection against downlink failure at a member RBridge; and Section 9 gives the necessary TRILL code points and data structures for a pseudo-nickname AAE RBridge group.

1.1. Terminology and Acronyms

This document uses the acronyms and terms defined in [RFC6325] and [RFC7379] and the following additional acronyms:

AAE - Active-active Edge RBridge group, a group of edge RBridges to which at least one CE is multiply attached with an LAALP. AAE is also referred to as edge group or Virtual RBridge in this document.

Campus - A TRILL network consisting of TRILL switches, links, and possibly bridges bounded by end stations and IP routers. For TRILL, there is no "academic" implication in the name "campus".

CE - Customer Equipment (end station or bridge). The device can be either physical or virtual equipment.

Data Label - VLAN or FGL.

DF - Designated Forwarder.

DRNI: Distributed Resilient Network Interconnect. A link aggregation specified in [802.1AX] that can provide an LAALP between from 1 to 3 CEs and 2 or 3 RBridges.

E-L1FS - Extended Level 1 Flooding Scope [RFC7356].

FGL - Fine-Grained Labeling or Fine-Grained Labeled or Fine-Grained Label [RFC7172].

LAALP - Local Active-Active Link Protocol [RFC7379] such as MC-LAG or DRNI.

MC-LAG: Multi-Chassis LAG. Proprietary extensions of Link Aggregation [802.1AX] that can provide an LAALP between one CE and 2 or more RBridges.

OE flag - A flag used by the member RBridge of an LAALP to tell other edge RBridges whether it is willing to share an RBv with other LAALPs if they multiply attach to the same set of edge RBridges as it. When this flag for an LAALP is 1, it means that the LAALP needs to be served by an RBv by itself and is not willing to share, that is, it should Occupy an RBv Exclusively (OE).

RBv - virtual RBridge, an alias for active-active edge RBridge group in this document.

vDRB - The Designated RBridge in an RBv. It is responsible for deciding the pseudo-nickname for the RBv.

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 RFC 2119 [RFC2119].

2. Overview

To minimize impact during failures and maximize available access bandwidth, Customer Equipment (referred to as CE in this document) may be multiply connected to TRILL campus via multiple edge RBridges.

Figure 1 shows such a typical deployment scenario, where CE1 attaches to RB1, RB2, ... RBk and treats all of the uplinks as an LAALP bundle. Then RB1, RB2, ... RBk constitute an Active-active Edge (AAE) RBridge group for CE1 in this LAALP. Even if a member RBridge or an uplink fails, CE1 will still get frame forwarding service from the TRILL campus if there are still member RBridges and uplinks available in the AAE group. Furthermore, CE1 can make flow-based load balancing across the available member links of the LAALP bundle in the AAE group when it communicates with other CEs across the TRILL campus [RFC7379].

```
                  ----------------------
                 |                      |
                 |     TRILL Campus     |
                 |                      |
                  ----------------------
                      |       |    |
                +-----+       |    +--------+
                |             |             |
            +------+      +------+      +------+
            |(RB1) |      |(RB2) |      | (RBk)|
            +------+      +------+      +------+
              |..|          |..|          |..|
              |  +----+     |  |          |  |
              |   +---|-----|--|----------+  |
              | +-|---|-----+  +-----------+ |
              | | |   +------------------+ | |
    LAALP1-->(| | |)                    (| | |) <--LAALPn
            +-------+    .  .  .       +-------+
            | CE1   |                  | CEn   |
            +-------+                  +-------+
```

Figure 1 Active-Active Connection to TRILL Edge RBridges

By design, an LAALP (say LAALP1) does not forward packets received on one member port to other member ports. As a result, the TRILL Hello messages sent by one member RBridge (say RB1) via a port to CE1 will not be forwarded to other member RBridges by CE1. That is to say, member RBridges will not see each other's Hellos via the LAALP. So every member RBridge of LAALP1 thinks of itself as appointed forwarder for all VLANs enabled on an LAALP1 link and can ingress/egress frames simultaneously in these VLANs [RFC6439].

The simultaneous flow-based ingressing/egressing can cause some problems. For example, simultaneous egressing of multi-destination traffic by multiple member RBridges will result in frame duplication at CE1 (see Section 3.1 of [RFC7379]); simultaneous ingressing of frames originated by CE1 for different flows in the same VLAN with the same source MAC address will result in MAC address flip-flopping at remote egress RBridges that have data plane address learning enabled (see Section 3.3 of [RFC7379]). The flip-flopping would in turn cause packet re-ordering in reverse traffic.

Edge RBridges learn Data Label and MAC address to nickname correspondences by default via decapsulating TRILL data packets (see Section 4.8.1 of [RFC6325] as updated by [RFC7172]). The MAC flip-flopping issue is solved herein based on the assumption that the default learning is enabled at edge RBridges, so this document specifies using a Virtual RBridge together with its pseudo-nickname.

3. Virtual RBridge and its Pseudo-nickname

A Virtual RBridge (RBv) represents a group of edge RBridges to which at least one CE is multiply attached using an LAALP. More exactly, it represents a group of ports on the edge RBridges providing end station service and the service provided to the CE(s) on these ports, through which the CE(s) are multiply attached to the TRILL campus using LAALP(s). Such end station service ports are called RBv ports; in contrast, other access ports at edge RBridges are called regular access ports in this document. RBv ports are always LAALP connecting ports, but not vice versa (see Section 4.1). For an edge RBridge, if one or more of its end station service ports are ports of an RBv, that RBridge is a member RBridge of that RBv.

For the convenience of description, a Virtual RBridge is also referred to as an Active-Active Edge (AAE) group in this document. In the TRILL campus, an RBv is identified by its pseudo-nickname, which is different from any RBridge’s regular nickname(s). An RBv has one and only one pseudo-nickname. Each member RBridge (say RB1, RB2 ..., RBk) of an RBv (say RBvn) advertises RBvn’s pseudo-nickname using a Nickname sub-TLV in its TRILL IS-IS LSP (Link State PDU) [RFC7176] and SHOULD do so with maximum priority of use (0xFF), along with their regular nickname(s). (Maximum priority is recommended to avoid the disruption to an AAE group that would occur if the nickname were taken away by a higher priority RBridge.) Then, from these LSPs, other RBridges outside the AAE group know that RBvn is reachable through RB1 to RBk.

A member RBridge (say RBi) loses its membership in RBvn when its last port in RBvn becomes unavailable due to failure, re-configuration, etc. Then RBi removes RBvn’s pseudo-nickname from its LSP and distributes the updated LSP as usual. From those updated LSPs, other RBridges know that there is no path to RBvn through RBi now.

When member RBridges receive native frames on their RBv ports and decide to ingress the frames into the TRILL campus, they use that RBv’s pseudo-nickname instead of their own regular nicknames as the ingress nickname to encapsulate them into TRILL Data packets. So when these packets arrive at an egress RBridge, even if they are originated by the same end station in the same VLAN but ingressed by different member RBridges, no address flip-flopping is observed on the egress RBridge when decapsulating these packets. (When a member RBridge of an AAE group ingresses a frame from a non-RBv port, it still uses its own regular nickname as the ingress nickname.)

Since RBv is not a physical node and no TRILL frames are forwarded between its ports via an LAALP, pseudo-node LSP(s) MUST NOT be created for an RBv. RBv cannot act as a root when constructing distribution trees for multi-destination traffic and its pseudo-nickname is ignored when determining the distribution tree root for TRILL campus [CMT]. So the tree root priority of RBv's nickname MUST be set to 0, and this nickname SHOULD NOT be listed in the "s" nicknames (see Section 2.5 of [RFC6325]) by the RBridge holding the highest priority tree root nickname.

NOTE: In order to reduce the consumption of nicknames, especially in large TRILL campus with lots of RBridges and/or active-active accesses, when multiple CEs attach to the exact same set of edge RBridges via LAALPs, those edge RBridges should be considered as a single RBv with a single pseudo-nickname.

4. Member RBridges Auto-Discovery

Edge RBridges connected to a CE via an LAALP can automatically discover each other with minimal configuration through exchange of LAALP connection information.

From the perspective of edge RBridges, a CE that connects to edge RBridges via an LAALP can be identified by the ID of the LAALP that is unique across the TRILL campus (for example, the MC-LAG or DRNI System ID [802.1AX]), which is referred to as an LAALP ID in this document. On each of such edge RBridges, the access port to such a CE is associated with an LAALP ID for the CE. An LAALP is considered valid on an edge RBridge only if the RBridge still has an operational down-link to that LAALP. For such an edge RBridge, it advertises a list of LAALP IDs for its valid local LAALPs to other edge RBridges via its E-L1FS FS-LSP(s) [RFC7356][rfc7180bis]. Based on the LAALP IDs advertised by other RBridges, each RBridge can know which edge RBridges could constitute an AAE group (See Section 4.1 for more details). Then one RBridge is elected from the group to allocate an available nickname (the pseudo-nickname) for the group (See Section 4.2 for more details).

4.1. Discovering Member RBridge for an RBv

Take Figure 2 as an example, where CE1 and CE2 multiply attach to RB1, RB2 and RB3 via LAALP1 and LAALP2 respectively; CE3 and CE4 attach to RB3 and RB4 via LAALP3 and LAALP4 respectively. Assume LAALP3 is configured to occupy a Virtual RBridge by itself.

```
                      ------------------------
                    /                          \
                   |       TRILL Campus         |
                    \                          /
                      -------------------------
                       |    |             |  |
               +-------+    |             |  +----------+
               |            |             |             |
           +-------+     +-------+      +-------+     +-------+
           |  RB1  |     |  RB2  |      |  RB3  |     |  RB4  |
           +-------+     +-------+      +-------+     +-------+
             |   |        |   |          | | | |       |     |
             |   +--------|--+ | +-------|-+ | +-------|---+ |
             | +----------+  | | |       |   |         |   | |
             | | +-----------|-|-|-------+   | +-------+   | |
             | | |           | | |           | |           | |
    LAALP1->(| | |) LAALP2->(| | |) LAALP3->(| |) LAALP4->(| |)
           +-------+      +-------+     +-------+      +-------+
           |  CE1  |      |  CE2  |     |  CE3  |      |  CE4  |
           +-------+      +-------+     +-------+      +-------+
```

Figure 2 Different LAALPs to TRILL Campus

RB1 and RB2 advertise {LAALP1, LAALP2} in the PN-LAALP-Membership sub-TLV (see Section 9.1 for more details) via their TRILL E-L1FS LSPs respectively; RB3 announces {LAALP1, LAALP2, LAALP3, LAALP}; and RB4 announces {LAALP3, LAALP4}, respectively.

An edge RBridge is called an LAALP related RBridge if it has at least one LAALP configured on an access port. On receipt of the PN-LAALP-Membership sub-TLVs, RBn ignores them if it is not an LAALP related RBridge; otherwise, RBn SHOULD use the LAALP information contained in the sub-TLVs, along with its own PN-LAALP-Membership sub-TLVs to decide which RBv(s) it should join and which edge RBridges constitute each of such RBvs. Based on the information received, each of the 4 RBridges knows the following information:

LAALP ID	OE-flag	Set of edge RBridges
LAALP1	0	{RB1, RB2, RB3}
LAALP2	0	{RB1, RB2, RB3}
LAALP3	1	{RB3, RB4}
LAALP4	0	{RB3, RB4}

Where the OE-flag indicates whether an LAALP is willing to share an RBv with other LAALPs if they multiply attach to exact the same set of edge RBridges as it. For an LAALP (for example LAALP3), if its OE-flag is one, it means that LAALP3 does not want to share, so it MUST

Occupy an RBv Exclusively (OE). Support of OE is optional. RBridges that do not support OE ignore the OE bit and act as if it was zero (see Section 11 on Configuration Consistency).

Otherwise, the LAALP (for example LAALP1) will share an RBv with other LAALPs if possible. By default, this flag is set to zero. For an LAALP, this flag is considered 1 if any edge RBridge advertises it as one (see Section 9.1).

In the above table, there might be some LAALPs that attach to a single RBridge due to mis-configuration or link failure, etc. Those LAALPs are considered as invalid entries. Then each of the LAALP related edge RBridges performs the following algorithm to decide which valid LAALPs can be served by an RBv.

Step 1: Take all the valid LAALPs that have their OE-flags set to 1 out of the table and create an RBv per such LAALP.

Step 2: Sort the valid LAALPs left in the table in descending order based on the number of RBridges in their associated set of multi-homed RBridges. In the case that several LAALPs have same number of RBridges, these LAALPs are then ordered in ascending order in the proper places of the table based on their LAALP IDs considered as unsigned integers. (for example, in the above table, both LAALP1 and LAALP2 have 3 member RBridges, assuming LAALP1 ID is smaller than LAALP2 ID, so LAALP1 is followed by LAALP2 in the ordered table.)

Step 3: Take the first valid LAALP (say LAALP_i) with the maximum set of RBridges, say S_i, out of the table and create a new RBv (Say RBv_i) for it.

Step 4: Walk through the remaining valid LAALPs in the table one by one, pick up all the valid LAALPs that have their sets of multi-homed RBridges contain exactly the same RBridges as that of LAALP_i and take them out of the table. Then appoint RBv_i as the servicing RBv for those LAALPs.

Step 5: Repeat Step 3-4 for any LAALPs left until all the valid entries in the table are associated with an RBv.

After performing the above steps, all the 4 RBridges know that LAALP3 is served by an RBv, say RBv1, which has RB3 and RB4 as member RBrdges; LAALP1 and LAALP2 are served by another RBv, say RBv2, which has RB1, RB2 and RB3 as member RBridges; and LAALP4 is served by RBv3, which has RB3 and RB4 as member RBridges, shown as follows:

RBv	Serving LAALPs	Member RBridges
RBv1	{LAALP3}	{RB3, RB4}
RBv2	{LAALP1, LAALP2}	{RB1, RB2, RB3}
RBv3	{LAALP4}	{RB3, RB4}

In each RBv, one of the member RBridges is elected as the vDRB (Designated RBridge) of the RBv. Then this RBridge picks up an available nickname as the pseudo-nickname for the RBv and announces it to all other member RBridges of the RBv via its TRILL E-L1FS LSPs (refer to Section 9.2 for the relative extended sub-TLVs).

4.2. Selection of Pseudo-nickname for RBv

As described in Section 3, in the TRILL campus, an RBv is identified by its pseudo-nickname. In an AAE group (i.e., RBv), one member RBridge is elected for the duty to select a pseudo-nickname for this RBv; this RBridge is called Designated RBridge of the RBv (vDRB) in this document. The winner is the RBridge with the largest IS-IS System ID considered as an unsigned integer, in the group. Then based on its TRILL IS-IS link state database and the potential pseudo-nickname(s) reported in the PN-LAALP-Membership sub-TLVs by other member RBridges of this RBv (see Section 9.1 for more details), the vDRB selects an available nickname as the pseudo-nickname for this RBv and advertizes it to the other RBridges via its E-L1FS FS-LSP(s) (see Section 9.2 and [rfc7180bis]). Except as provided below, the selection of a nickname to use as the pseudo-nickname follows the usual TRILL rules given in [RFC6325] as updated by [rfc7180bis]. On receipt of the pseudo-nickname advertised by the vDRB, all the other RBridges of that group associate it with the LAALPs served by the RBv, and then download the association to their data plane fast path logic.

To reduce the traffic disruption caused by nickname changing, if possible, vDRB SHOULD attempt to reuse the pseudo-nickname recently used by the group when selecting nickname for the RBv. To help the vDRB to do so, each LAALP related RBridge advertises a re-using pseudo-nickname for each of its LAALPs in its LAALP Membership sub-TLV if it has used such a pseudo-nickname for that LAALP recently. Although it is up to the implementation of the vDRB as to how to treat the re-using pseudo-nicknames, the following is RECOMMENDED:

o If there are multiple available re-using pseudo-nicknames that are reported by all the member RBridges of some LAALPs in this RBv, the available one that is reported by the largest number of such LAALPs is chosen as the pseudo-nickname for this RBv. If a tie exists, the re-using pseudo-nickname with the smallest value considered as an unsigned integer is chosen.

- o If only one re-using pseudo-nickname is reported, it SHOULD be chosen if available.

If there is no available re-using pseudo-nickname reported, the vDRB selects a nickname by its usual method.

Then the selected pseudo-nickname is announced by the vDRB to other member RBridges of this RBv in the PN-RBv sub-TLV (see Section 9.2). After receiving the pseudo-nickname, other RBridges of that RBv associate the nickname with their ports of that RBv and download the association to their data plane fast path logic.

5. Distribution Trees and Designated Forwarder

In an AAE group (i.e., an RBv), as each of the member RBridges thinks it is the appointed forwarder for VLAN x, without changes made for active-active connection support, they would all ingress/egress frames into/from TRILL campus for all VLANs. For multi-destination frames, more than one member RBridges ingressing them may cause some of the resulting TRILL Data packets to be discarded due to failure of Reverse Path Forwarding (RPF) Check on other RBridges; for a multi-destination traffic, more than one RBridges egressing it may cause local CE(s) receiving duplication frame. Furthermore, in an AAE group, a multi-destination frame sent by a CE (say CEi) may be ingressed into TRILL campus by one member RBridge, then another member RBridge will receive it from TRILL campus and egress it to CEi, which will result in loop back of frame for CEi. These problems are all described in [RFC7379].

In the following sub-sections, the first two issues are discussed in Section 5.1 and Section 5.2, respectively; the third one is discussed in Section 5.3.

5.1. Different Trees for Different Member RBridges

In TRILL, RBridges normally use distribution trees to forward multi-destination frames. (Under some circumstances they can be unicast as specified in [RFC7172].) An RPF Check along with other checking is used to avoid temporary multicast loops during topology changes (Section 4.5.2 of [RFC6325]). The RPF check mechanism only accepts a multi-destination frame ingressed by an RBridge RBi and forwarded on a distribution tree Tx if it arrives at another RBridge RBn on the expected port. If arriving on any other port, the frame MUST be dropped.

To avoid address flip-flopping on remote RBridges, member RBridges use RBv’s pseudo-nickname instead of their regular nicknames as ingress nickname to ingress native frames, including multi-destination frames. From the view of other RBridges, these frames appear as if they were ingressed by the RBv. When multi-destination frames of different flows are ingressed by different member RBridges of an RBv and forwarded along the same distribution tree, they may arrive at RBn on different ports. Some of them will violate the RPF check principle at RBn and be dropped, which will result in lost traffic.

In an RBv, if different member RBridge uses different distribution trees to ingress multi-destination frames, the RPF check violation issue can be fixed. Coordinated Multicast Trees (CMT) proposes such an approach, and makes use of the Affinity sub-TLV defined in [RFC7176] to tell other RBridges which trees a member RBridge (say RBi) may choose when ingressing multi-destination frames;then all RBridges in the TRILL campus can calculate RPF check information for RBi on those trees taking the tree affinity information into account [CMT].

This document uses the approach proposed in [CMT] to fix the RPF check violation issue. Please refer to [CMT] for more details of the approach. An alternative solution is proposed in [CentralReplicate].

5.2. Designated Forwarder for Member RBridges

Take Figure 3 as an example, where CE1 and CE2 are served by an RBv that has RB1 and RB2 as member RBridges. In VLAN x, the three CEs can communicate with each other.

```
                    ---------------------
                  /                       \    +-----+
                 |       TRILL Campus      |---| RBn |
                  \                       /    +-----+
                   -----------------------
                       |             |
                  +----+             +------+
                  |                         |
             +---------+               +---------+
             |   RB1   |               |   RB2   |
             | oooooooo|oooooooooooooooo|ooooo   |
             +o--------+    RBv        +-----o--+
               o|oooo|ooooooooooooooooooo|o|o   |
                | +--|-------------------+ |    |
                | |  +---------+ +----------+    |
               (| |)<-LAALP1  (| |)<-LAALP2     |
            +-------+       +-------+      +-------+
            |  CE1  |       |  CE2  |      |  CE3  |
            +-------+       +-------+      +-------+
```

Figure 3 A Topology with Multi-homed and Single-homed CEs

When a remote RBridge (say RBn) sends a multi-destination TRILL Data packet in VLAN x (or the FGL that VLAN x maps to if the packet is FGL), both RB1 and RB2 will receive it. As each of them thinks it is the appointed forwarder for VLAN x, without changes made for active-active connection support, they would both forward the frame to CE1/CE2. As a result, CE1/CE2 would receive duplicate copies of the frame through this RBv.

In another case, assume CE3 is single-homed to RB2. When it transmits a native multi-destination frame onto link CE3-RB2 in VLAN x, the frame can be locally replicated to the ports to CE1/CE2, and also encapsulated into TRILL Data packet and ingressed into TRILL campus. When the packet arrives at RB1 across the TRILL campus, it will be egressed to CE1/CE2 by RB1. Then CE1/CE2 receives duplicate copies from RB1 and RB2.

In this document, the Designated Forwarder (DF) for a VLAN is introduced to avoid the duplicate copies. The basic idea of DF is to elect one RBridge per VLAN from an RBv to egress multi-destination TRILL Data traffic and replicate locally-received multi-destination native frames to the CEs served by the RBv.

Note that DF has an effect only on the egressing/replicating of multi-destination traffic, no effect on the ingressing of frames or forwarding/egressing of unicast frames. Furthermore, the DF check is performed only for RBv ports, not on regular access ports.

Each RBridge in an RBv elects a DF using the same algorithm which guarantees the same RBridge elected as DF per VLAN by all members of the RBv.

Assuming there are m LAALPs and k member RBridges in an RBv; each LAALP is referred to as LAALPi where 0 <= i < m, and each RBridge is referred to as RBj where 0 <= j < k-1, the DF election algorithm per VLAN is as follows:

Step 1: For LAALPi, sort all the RBridges in numerically ascending order based on (System IDj | LAALPi) mod k, where "System IDj" is the IS-IS System ID of RBj, "|" means concatenation, and LAALPi is the LAALP ID for LAALPi. In the case that some RBridges get the same result of the mod operation, those RBridges are sorted in numerically ascending order by their System IDs considered as unsigned integers.

Step 2: Each RBridge in the numerically sorted list is assigned a monotonically increasing number j, such that increasing number j corresponds to its position in the sorted list, i.e., the first RBridge (the first one with the smallest (System ID | LAALP ID) mod k) is assigned zero and the last is assigned k-1.

Step 3: For each VLAN ID n, choose the RBridge whose number equals (n mod k) as the DF.

Step 4: Repeat Step 1-3 for the remaining LAALPs until there is a DF per VLAN per LAALP in the RBv.

For a multi-destination native frame of VLAN x received, if RBi is an LAALP attached RBridge, in addition to local replication of the frame to regular access ports as per [RFC6325] (and [RFC7172] for FGL), it MUST also locally replicate the frame to the following RBv ports when one of the following conditions is met:

1) RBv ports associated with the same pseudo-nickname as that of the incoming port, no matter whether RBi is the DF for the frame’s VLAN on the outgoing ports except that the frame MUST NOT be replicated back to the incoming port;

2) RBv ports on which RBi is the DF for the frame’s VLAN while they are associated with different pseudo-nickname(s) to that of the incoming port.

For non-LAALP related RBridges or for non-RBv ports on an LAALP related RBridge, local replication is performed as per [RFC6325].

For a multi-destination TRILL Data packet received, RBi MUST NOT egress it out of the RBv ports where it is not DF for the frame’s

Inner.VLAN (or for the VLAN corresponding to the Inner.Label if the packet is an FGL one). Otherwise, whether or not egressing it out of such ports is further subject to the filtering check result of the frame's ingress nickname on these ports (see Section 5.3).

5.3. Ingress Nickname Filtering

As shown in Figure 3, CE1 may send multi-destination traffic in VLAN x to TRILL campus via a member RBridge (say RB1). The traffic is then TRILL-encapsulated by RB1 and delivered through the TRILL campus to multi-destination receivers. RB2 may receive the traffic, and egress it back to CE1 if it is the DF for VLAN x on the port to LAALP1. Then the traffic loops back to CE1 (see Section 3.2 of [RFC7379).

To fix the above issue, an ingress nickname filtering check is required by this document. The idea of this check is to check the ingress nickname of a multi-destination TRILL Data packet before egressing a copy of it out of an RBv port. If the ingress nickname matches the pseudo-nickname of the RBv (associated with the port), the filtering check should fail and the copy MUST NOT be egressed out of that RBv port. Otherwise, the copy is egressed out of that port if it has also passed other checks, such as the appointed forwarder check in Section 4.6.2.5 of [RFC6325] and the DF check in Section 5.2.

Note that this ingress nickname filtering check has no effect on the multi-destination native frames received on access ports and replicated to other local ports (including RBv ports), since there is no ingress nickname associated with such frames. Furthermore, for the RBridge regular access ports, there is no pseudo-nickname associated with them; so no ingress nickname filtering check is required on those ports.

More details of data packet processing on RBv ports are given in the next section.

6. TRILL Traffic Processing

This section provides more details of native frame and TRILL Data packet processing as it relates to the RBv's pseudo-nickname.

6.1. Native Frames Ingressing

When RB1 receives a unicast native frame from one of its ports that has end-station service enabled, it processes the frame as described in Section 4.6.1.1 of [RFC6325] with the following exception.

o If the port is an RBv port, RB1 uses the RBv’s pseudo-nickname, instead of one of its regular nickname(s) as the ingress nickname when doing TRILL encapsulation on the frame.

When RB1 receives a native multi-destination (Broadcast, Unknown unicast or Multicast) frame from one of its access ports (including regular access ports and RBv ports), it processes the frame as described in Section 4.6.1.2 of [RFC6325] with the following exceptions.

o If the incoming port is an RBv port, RB1 uses the RBv’s pseudo-nickname, instead of one of its regular nickname(s) as the ingress nickname when doing TRILL encapsulation on the frame.

o For the copies of the frame replicated locally to RBv ports, there are two cases as follows:

  - If the outgoing port(s) is associated with the same pseudo-nickname as that of the incoming port but not with the same LAALP as the incoming port, the copies are forwarded out of that outgoing port(s) after passing the appointed forwarder check for the frame’s VLAN. That is to say, the copies are processed on such port(s) as Section 4.6.1.2 of [RFC6325].

  - Else, the Designated Forwarder (DF) check is also made on the outgoing ports for the frame’s VLAN after the appointed forwarder check. The copies are not output through the ports that failed the DF check (i.e., RB1 is not DF for the frame’s VLAN on the ports); otherwise, the copies are forwarded out of the ports that pass the DF check (see Section 5.2).

For such a frame received, the MAC address information learned by observing it, together with the LAALP ID of the incoming port SHOULD be shared with other member RBridges in the group (see Section 7).

6.2. Egressing TRILL Data Packets

This section describes egress processing of the TRILL Data packets received on an RBv member RBridge (say RBn). Section 6.2.1 describes the egress processing of unicast TRILL Data packets and Section 6.2.2 specifies the multi-destination TRILL Data packets egressing.

6.2.1. Unicast TRILL Data Packets

When receiving a unicast TRILL data packet, RBn checks the egress nickname in the TRILL header of the packet. If the egress nickname is one of RBn’s regular nicknames, the packet is processed as defined in Section 4.6.2.4 of [RFC6325].

If the egress nickname is the pseudo-nickname of a local RBv, RBn is responsible for learning the source MAC address, unless data plane learning has been disabled. The learned {Inner.MacSA, Data Label, ingress nickname} triplet SHOULD be shared within the AAE group as described in Section 7.

Then the packet is de-capsulated to its native form. The Inner.MacDA and Data Label are looked up in RBn’s local forwarding tables, and one of the three following cases will occur. RBn uses the first case that applies and ignores the remaining cases:

- o If the destination end station identified by the Inner.MacDA and Data Label is on a local link, the native frame is sent onto that link with the VLAN from the Inner.VLAN or VLAN corresponding to the Inner.Label if the packet is FGL.

- o Else if RBn can reach the destination through another member RBridge RBk, it tunnels the native frame to RBk by re-encapsulating it into a unicast TRILL Data packet and sends it to RBk. RBn uses RBk’s regular nickname, instead of the pseudo-nickname as the egress nickname for the re-encapsulation, and the ingress nickname remains unchanged (somewhat similar to Section 2.4.2.1 of [rfc7180bis]). If the hop count value of the packet is too small for it to reach RBk safely, RBn SHOULD increase that value properly in doing the re-encapsulation. (NOTE: When receiving that re-encapsulated TRILL Data packet, as the egress nickname of the packet is RBk’s regular nickname rather than the pseudo-nickname of a local RBv, RBk will process it as Section 4.6.2.4 of [RFC6325], and will not re-forward it to another RBridge.)

- o Else, RBn does not know how to reach the destination; it sends the native frame out of all the local ports on which it is appointed forwarder for the Inner.VLAN (or appointed forwarder for the VLAN into which the Inner.Label maps on that port for FGL TRILL Data packet [RFC7172]).

6.2.2. Multi-Destination TRILL Data Packets

When RB1 receives a multi-destination TRILL Data Packet, it checks and processes the packet as described in Section 4.6.2.5 of [RFC6325] with the following exception.

- o On each RBv port where RBn is the appointed forwarder for the packet’s Inner.VLAN (or for the VLAN to which the packet’s Inner.Label maps on that port if it is an FGL TRILL Data packet), the Designated Forwarder check (see Section 5.2) and the Ingress Nickname Filtering check (see Section 5.3) are further performed.

For such an RBv port, if either the DF check or the filtering check fails, the frame MUST NOT be egressed out of that port. Otherwise, it can be egressed out of that port.

7. MAC Information Synchronization in Edge Group

An edge RBridge, say RB1 in LAALP1, may have learned a { MAC address, Data Label } to nickname correspondence for a remote host h1 when h1 sends a packet to CE1. The returning traffic from CE1 may go to another member RBridge of LAALP1, for example RB2. RB2 may not have that correspondence stored. Therefore it has to do the flooding for unknown unicast. Such flooding is unnecessary since the returning traffic is almost always expected and RB1 had learned the address correspondence. To avoid the unnecessary flooding, RB1 SHOULD share the correspondence with other RBridges of LAALP1. RB1 synchronizes the correspondence by using the MAC-RI sub-TLV [RFC6165] in its ESADI-LSPs [RFC7357].

On the other hand, RB2 has learned the MAC address and Data Label of CE1 when CE1 sends a frame to h1 through RB2. The returning traffic from h1 may go to RB1. RB1 may not have CE1's MAC address and Data Label stored even though it is in the same LAALP for CE1 as RB2. Therefore it has to flood the traffic out of all its access ports where it is appointed forwarder for the VLAN (see Section 6.2.1) or the VLAN the FGL maps to on that port if the packet is FGL. Such flooding is unnecessary since the returning traffic is almost always expected and RB2 had learned the CE1's MAC and Data Label information. To avoid that unnecessary flooding, RB2 SHOULD share the MAC address and Data Label with other RBridges of LAALP1. RB2 synchronizes the MAC address and Data Label by enclosing the relative MAC-RI TLV within a pair of boundary TRILL APPsub-TLVs for LAALP1 (see Section 9.3) in its ESADI-LSP [RFC7357]. After receiving the enclosed MAC-RI TLVs, the member RBridges of LAALP1 (i.e., LAALP1 related RBridges) treat the MAC address and Data Label as if it was learned by them locally on their member port of LAALP1; the LAALP1 unrelated RBridges just ignore LAALP1's boundary APPsub-TLVs and treat the MAC address and Data Label as specified in [RFC7357]. Furthermore, in order to make the LAALP1 unrelated RBridges know that the MAC and Data Label is reachable through the RBv that provides service to LAALP1, the Topology-id/Nickname field of the MAC-RI TLV SHOULD carry the pseudo-nickname of the RBv rather than zero or one of the originating RBridge's (i.e., RB2's) regular nicknames.

8. Member Link Failure in RBv

As shown in Figure 4, suppose the link RB1-CE1 fails. Although a new

RBv will be formed by RB2 and RB3 to provide active-active service for LAALP1 (see Section 5), the unicast traffic to CE1 might still be forwarded to RB1 before the remote RBridge learns CE1 is attached to the new RBv. That traffic might be disrupted by the link failure. Section 8.1 discusses the failure protection in this scenario.

However, for multi-destination TRILL Data packets, since they can reach all member RBridges of the new RBv and be egressed to CE1 by either RB2 or RB3 (i.e., the new DF for the traffic's Inner.VLAN or the VLAN the packet's Inner.Label maps to in the new RBv), special actions to protect against down-link failure for such multi-desination packets is not needed.

```
                   ------------------
                 /                    \
                |     TRILL Campus     |
                 \                    /
                  --------------------
                      |     |     |
                  +---+     |     +----+
                  |         |          |
              +------+     +------+   +------+
              | RB1  |     | RB2  |   | RB3  |
              ooooooo|ooooo|oooooo|ooo|ooooo |
             o+------+ RBv +------+   +-----o+
              o|oooo|ooooooo|oooo|ooooo|oo|o
               |    |       |  +-|-----+  |
              \|/+--|-------+  | +------+ |
             - B |  +----------|------+ | |
              /|\| +-----------+      | | |
              (| | |)<--LAALP1        (| | |)<--LAALP2
             +-------+              +-------+
             |  CE1  |              |  CE2  |
             +-------+              +-------+
```

B - Failed Link or Link bundle

Figure 4 A Topology with Multi-homed and Single-homed CEs

8.1. Link Protection for Unicast Frame Egressing

When the link CE1-RB1 fails, RB1 loses its direct connection to CE1. The MAC entry through the failed link to CE1 is removed from RB1's local forwarding table immediately. Another MAC entry learned from another member RBridge of LAALP1 (for example RB2, since it is still a member RBridge of LAALP1) is installed into RB1's forwarding table (see Section 9.3). In that new entry, RB2 (identified by one of its regular nicknames) is the egress RBridge for CE1's MAC address. Then when a TRILL Data packet to CE1 is delivered to RB1, it can be tunneled to RB2 after being re-encapsulated (ingress nickname remains unchanged and egress nickname is replaced by RB2's regular nickname) based on the above installed MAC entry (see bullet 2 in Section 6.2.1). Then RB2 receives the frame and egresses it to CE1.

After the failure recovery, RB1 learns that it can reach CE1 via link CE1-RB1 again by observing CE1's native frames or from the MAC information synchronization by member RBridge(s) of LAALP1 described in Section 7, then it restores the MAC entry to its previous one and downloads it to its data plane fast path logic.

9. TLV Extensions for Edge RBridge Group

9.1. PN-LAALP-Membership APPsub-TLV

This APPsub-TLV is used by an edge RBridge to announce its associated pseudo-nickname LAALP information. It is defined as a sub-TLV of the TRILL GENINFO TLV [RFC7357] and is distributed in E-L1FS FS-LSPs [rfc7180bis]. It has the following format:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Type = PN-LAALP-Membership   |  (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Length                       |  (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...+-+
|  LAALP RECORD(1)                              |  (variable)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...+-+
.                                               .
.                                               .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...+-+
|  LAALP RECORD(n)                              |  (variable)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...+-+
```

Figure 5 PN-LAALP-Membership Advertisement APPsub-TLV

where each LAALP RECORD has the following form:

```
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 ..
+--+-+-+-+-+-+-+-+-+
|OE|     RESV    |                  (1 byte)
+--+-+-+-+-+-+-+-+-+
|  Size          |                  (1 byte)
+--+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Re-using Pseudo-nickname      |  (2 bytes)
+--+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...+-+
|  LAALP ID                                     |  (variable)
+--+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...+-+
```

o PN-LAALP-Membership (2 bytes): Defines the type of this sub-TLV, #tbd1.

o Length (2 bytes): the sum of the lengths of the LAALP RECORDs.

o OE (1 bit): a flag indicating whether or not the LAALP wants to occupy an RBv by itself; 1 for occupying by itself (or Occupying Exclusively (OE)). By default, it is set to 0 on transmit. This bit is used for edge RBridge group auto-discovery (see Section 4.1). For any one LAALP, the values of this flag might conflict in the LSPs advertised by different member RBridges of that LAALP. In that case, the flag for that LAALP is considered as 1.

o RESV (7 bits): MUST be transmitted as zero and ignored on receipt.

o Size (1 byte): Size of remaining part of LAALP RECORD (2 plus length of the LAALP ID).

o Re-using Pseudo-nickname (2 bytes): Suggested pseudo-nickname of the AAE group serving the LAALP. If the LAALP is not served by any AAE group, this field MUST be set to zero. It is used by the originating RBridge to help the vDRB to reuse the previous pseudo-nickname of an AAE group (see Section 4.2).

o LAALP ID (variable): The ID of the LAALP. If the LAALP is an MC-LAG or DRNI, it is the 8 byte ID as specified in Section 6.3.2 in [802.1AX].

On receipt of such an APPsub-TLV, if RBn is not an LAALP related edge RBridge, it ignores the sub-TLV; otherwise, it parses the sub-TLV. When new LAALPs are found or old ones are withdrawn compared to its old copy, and they are also configured on RBn, it triggers RBn to perform the "Member RBridges Auto-Discovery" procedure described in Section 4.1.

9.2. PN-RBv APPsub-TLV

The PN-RBv APPsub-TLV is used by a Designated RBridge of a Virtual RBridge (vDRB) to dictate the pseudo-nickname for the LAALPs served by the RBv. It is defined as a sub-TLV of TRILL GENINFO TLV [RFC7357] and is distributed in E-L1FS FS-LSP [rfc7180bis]. It has the following format:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Type = PN-RBv                 |  (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Length                        |  (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| RBv's Pseudo-Nickname         |  (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| LAALP ID Size |  (1 byte)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...+-+
| LAALP ID (1)                                  |  (variable)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...+-+
.                                               .
.                                               .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...+-+
| LAALP ID (n)                                  |  (variable)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...+-+
```

- o PN-RBv (2 bytes): Defines the type of this sub-TLV, #tbd2.

- o Length (2 bytes): 3+n*k bytes, where there are n LAALP IDs, each of size k bytes. k is found in the LLALP ID Size field below. If Length is not 3 plus an integer time k, the sub-TLV is corrupt and MUST be ignored.

- o RBv’s Pseudo-Nickname (2 bytes): The appointed pseudo-nickname for the RBv that serves for the LAALPs listed in the following fields.

- o LAALP ID Size (1 byte): The size of each of the following LAALP IDs in this sub-TLV. 8 if the LAALPs listed are MC-LAGs or DRNI (Section 6.3.2 in [802.1AX]). The value in this field is the k that appears in the formula for Length above.

- o LAALP ID (LAAP ID Size bytes): The ID of the LAALP.

This sub-TLV may occur multiple times with the same RBv pseudo-nickname with the meaning that all of the LAALPs listed are identified by that pseudo-nickname. For example, if there are LAALP IDs of different length, then the LAALP IDs of each size would have to be listed in a separate sub-TLV.

On receipt of such a sub-TLV, if RBn is not an LAALP related edge RBridge, it ignores the sub-TLV. Otherwise, if RBn is also a member RBridge of the RBv identified by the list of LAALPs, it associates the pseudo-nickname with the ports of these LAALPs and downloads the association to data plane fast path logic.

9.3. PN-MAC-RI-LAALP Boundary APPsub-TLVs

In this document, two APPsub-TLVs are used as boundary APPsub-TLVs for edge RBridge to enclose the MAC-RI TLV(s) containing the MAC address information leant form local port of an LAALP when this RBridge wants to share the information with other edge RBridges. They are defined as TRILL APPsub-TLVs [RFC7357]. The PN-MAC-RI-LAALP-INFO-START APPsub-TLV has the following format:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Type=PN-MAC-RI-LAALP-INFO-START| (2 byte)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Length                        | (2 byte)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...+-+-+-+-+-+-+-+
| LAALP ID                                 | (variable)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...+-+-+-+-+-+-+-+
```

o PN-MAC-RI-LAALP-INFO-START (2 bytes): Defines the type of this APPsub-TLV, #tbd3.

o Length (2 bytes): the size of the following LAALP ID. 8 if the LAALP listed is an MAC-LAG or DRNI.

o LAALP ID (variable): The ID of the LAALP (for example, for an MC-LAG or DRNI the ID as specified in Section 6.3.2 in [802.1AX]). This ID identifies the LAALP for all MAC addresses contained in following MAC-RI TLVs until a PN-MAC-RI-LAALP-INFO-END APPsub-TLV is encountered.

PN-MAC-RI-LAALP-INFO-END APPsub-TLV is defined as follows:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Type=PN-MAC-RI-LAALP-INFO-END | (2 byte)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Length                        | (2 byte)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

o PN-MAC-RI-LAALP-INFO-END (2 bytes): Defines the type of this sub-TLV, #tbd4.

o Length (2 bytes): 0.

This pair of APPsub-TLVs can be carried multiple times in an ESADI LSP and in multiple ESADI-LSPs. When an LAALP related edge RBridge (say RBn) wants to share with other edge RBridges the MAC addresses learned on its local ports of different LAALPs, it uses one or more pairs of such APPsub-TLVs for each of such LAALPs in its ESADI-LSPs. Each encloses the MAC-RI TLVs containing the MAC addresses learned from a specific LAALP. Furthermore, if the LAALP is served by a local RBv, the value of Topology ID/Nickname field in the relative MAC-RI

TLVs SHOULD be the pseudo-nickname of the RBv rather than one of the RBn's regular nickname or zero. Then on receipt of such a MAC-RI TLV, remote RBridges know that the contained MAC addresses are reachable through the RBv.

On receipt of such boundary APPsub-TLVs, when the edge RBridge is not an LAALP related one or cannot recognize such sub-TLVs, it ignores them and continues to parse the enclosed MAC-RI TLVs per [RFC7357]. Otherwise, the recipient parses the boundary APPsub-TLVs. The PN-MAC-RI-LAALP-INFO-START / PN-MAC-RI-LAALP-INFO-END pair MUST occur within one TRILL GENINFO TLV. If an END is encountered without any previous START in the ESADI-LSP, the END APPsub-TLV is ignored. If, after encountering a START, the end of the ESADI-LSP is reached without encountering an END, then the end of the ESADI-LSP is treated as if it were a PN-MAC-RI-LAALP-INFO-END. The boundary APPsub-TLVs and TLVs between them are handled as follows:

1) If the edge RBridge is configured with the contained LAALP and the LAALP is also enabled locally, it treats all the MAC addresses, contained in the following MC-RI TLVs enclosed by the corresponding pair of boundary APPsub-TLVs, as if they were learned from its local port of that LAALP;

2) Else, it ignores these boundary APPsub-TLVs and continues to parse the following MAC-RI TLVs per [RFC7357] until another pair of boundary APPsub-TLVs is encountered.

10. OAM Packets

Attention must be paid when generating OAM packets. To ensure the response messages can return to the originating member RBridge of an RBv, pseudo-nickname cannot be used as the ingress nickname in TRILL OAM messages, except in the response to an OAM message that has that RBv's pseudo-nickname as egress nickname. For example, assume RB1 is a member RBridge of RBvi, RB1 cannot use RBvi's pseudo-nickname as the ingress nickname when originating OAM messages; otherwise the responses to the messages may be delivered to another member RBridge of RBvi rather than RB1. But when RB1 responds to the OAM message with RBvi's pseudo-nickname as egress nickname, it can use that pseudo-nickname as the ingress nickname in the response message.

Since RBridges cannot use OAM messages for the learning of MAC addresses (Section 3.2.1 of [RFC7174]), it will not lead to MAC address flip-flopping at a remote RBridge even though RB1 uses its regular nicknames as ingress nicknames in its TRILL OAM messages while uses RBvi's pseudo-nickname in its TRILL Data packets.

11. Configuration Consistency

It is important that the VLAN membership of all the RBridge ports in an LAALP MUST be the same. Any inconsistencies in VLAN membership may result in packet loss or non-shortest paths.

Take Figure 1 for example, suppose RB1 configures VLAN1 and VLAN2 for the link CE1-RB1, while RB2 only configures VLAN1 for the CE1-RB2 link. Both RB1 and RB2 use the same ingress nickname RBv for all frames originating from CE1. Hence, a remote RBridge RBx will learn that CE1's MAC address in VLAN2 is originating from RBv. As a result, on the returning path, remote RBridge RBx may deliver VLAN2 traffic to RB2. However, RB2 does not have VLAN2 configured on CE1-RB2 link and hence the frame may be dropped or has to be redirected to RB1 if RB2 knows RB1 can reach CE1 in VLAN2.

It is important that if any VLAN in an LAALP is being mapped by edge RBridges to an FGL [RFC7172], that the mapping MUST be same for all edge RBridge ports in the LAALP. Otherwise, for example, unicast FGL TRILL Data packets from remote RBridges may get mapped into different VLANs depending on which edge RBridge receives and egresses them.

It is important that RBridges in an AAE group not be configured to assert the OE bit if any RBridge in the group does not implement it. Since, as stated in [RFC7379], the RBridges in an AAE edge group are expected to be from the same vendor, due to the proprietary nature of deployed LAALPs, this will normally follow automatically from all of the RBridge in an AAE edge group supporting or all not supporting OE.

12. Security Considerations

Authenticity for contents transported in IS-IS PDUs is enforced using regular IS-IS security mechanism [IS-IS] [RFC5310].

For security considerations pertain to extensions transported by TRILL ESADI, see the Security Considerations section in [RFC7357].

This draft does not introduce any extra security risks. For general TRILL Security Considerations, see [RFC6325].

13. IANA Considerations

IANA is requested to allocate code points tbd1, tbd2, tbd3 and tbd4 from the range below 255 for the 4 TRILL APPsub-TLVs specified in Section 9 and add them to the TRILL APPsub-TLV Types registry as follows:

Type	Name	Reference
tbd1	PN-LAALP-Membership	[this document]
tbd2	PN-RBv	[this document]
tbd3	PN-MAC-RI-LAALP-INFO-START	[this document]
tbd4	PN-MAC-RI-LAALP-INFO-END	[this document]

14. Acknowledgments

We would like to thank Mingjiang Chen for his contributions to this document. Additionally, we would like to thank Erik Nordmark, Les Ginsberg, Ayan Banerjee, Dinesh Dutt, Anoop Ghanwani, Janardhanan Pathang, Jon Hudson and Fangwei Hu for their good questions and comments.

15. Contributing Authors

Weiguo Hao
Huawei Technologies
101 Software Avenue,
Nanjing 210012
China

Phone: +86-25-56623144
Email: haoweiguo@huawei.com

Donald E. Eastlake, III
Huawei Technologies
155 Beaver Street
Milford, MA 01757 USA

Phone: +1-508-333-2270
Email: d3e3e3@gmail.com

16. References

16.1. Normative References

[CMT] T. Senevirathne, J. Pathangi, and J. Hudson, "Coordinated Multicast Trees (CMT) for TRILL", draft-ietf-trill-cmt Work in Progress.

[IS-IS] ISO/IEC 10589:2002, Second Edition, "Information technology -- Telecommunications and information exchange between systems -- 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)", 2002.

[RFC2119] S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC5310] M. Bhatia, V. Manral, T. Li, et al, "IS-IS Generic Cryptographic Authentication", RFC 5310, February 2009.

[RFC6165] Banerjee, A. and D. Ward, "Extensions to IS-IS for Layer-2 Systems", RFC 6165, April 2011.

[RFC6325] R. Perlman, D. Eastlake, D. Dutt, S. Gai, and A. Ghanwani, "Routing Bridges (RBridges): Base Protocol Specification", RFC 6325, July 2011.

[RFC6439] Perlman, R., Eastlake, D., Li, Y., Banerjee, A., and F. Hu, "Routing Bridges (RBridges): Appointed Forwarders", RFC 6439, November 2011, <http://www.rfc-editor.org/info/rfc6439>

[RFC7172] Eastlake 3rd, D., Zhang, M., Agarwal, P., Perlman, R., and D. Dutt, "Transparent Interconnection of Lots of Links (TRILL): Fine-Grained Labeling", RFC 7172, May 2014.

[RFC7176] D. Eastlake, A. Banerjee, A. Ghanwani, and R. Perlman, "Transparent Interconnection of Lots of Links (TRILL) Use of IS-IS", RFC 7176, May 2014.

[RFC7357] Zhai, H., Hu, F., Perlman, R., Eastlake 3rd, D., and O. Stokes, "Transparent Interconnection of Lots of Links (TRILL): End Station Address Distribution Information (ESADI) Protocol", RFC 7357, September 2014.

[RFC7356] Ginsberg, L., Previdi, S., and Y. Yang, "IS-IS Flooding Scope Link State PDUs (LSPs)", RFC 7356, September 2014.

[rfc7180bis] D. Eastlake, et al., draft-ietf-trill-frc7180bis, work in progress.

[802.1AX] IEEE, "IEEE Standard for Local and Metropolitan Area/ networks Link Aggregation", 802.1AX-2008, 1 January 2008.

16.2. Informative References

[802.1AX] IEEE, "IEEE Standard for Local and Metropolitan Area/networks Link Aggregation", 802.1AX-2014, 24 December 2014.

[RFC7174] Salam, S., Senevirathne, T., Aldrin, S., and D. Eastlake 3rd, "Transparent Interconnection of Lots of Links (TRILL) Operations, Administration, and Maintenance (OAM) Framework", RFC 7174, May 2014, <http://www.rfc-editor.org/info/rfc7174>.

[RFC7379] Li, Y., Hao, W., Perlman, R., Hudson, J., and H. Zhai, "Problem Statement and Goals for Active-Active Connection at the Transparent Interconnection of Lots of Links (TRILL) Edge", RFC 7379, October 2014.

[MultiAttach] Zhang, M., et al, "TRILL Active-Active Edge Using Multiple MAC Attachments", draft-ietf-trill-aa-multi-attach, Work in Progress.

[CentralReplicate] Hao, W., et al, "Centralized Replication for BUM traffic in active-active edge connection", draft-ietf-trill-centralized-replication, Work in Progress.

Authors' Addresses

Hongjun Zhai
Jinling Institute of Technology
99 Hongjing Avenue, Jiangning District
Nanjing, Jiangsu 211169
China

Email: honjun.zhai@tom.com

Tissa Senevirathne
Cisco Systems
375 East Tasman Drive
San Jose, CA 95134
USA

Phone: +1-408-853-2291
Email: tsenevir@cisco.com

Radia Perlman
EMC
2010 256th Avenue NE, #200
Bellevue, WA 98007
USA

Email: Radia@alum.mit.edu

Mingui Zhang
Huawei Technologies
Huawei Building, No.156 Beiqing Rd.
Beijing, Beijing 100095
China

Email: zhangmingui@huawei.com

Yizhou Li
Huawei Technologies
101 Software Avenue,
Nanjing 210012
China

Phone: +86-25-56625409
Email: liyizhou@huawei.com

TRILL Working Group — Radia Perlman, EMC
INTERNET-DRAFT — Donald Eastlake
Intended status: Informational — Mingui Zhang, Huawei
Anoop Ghanwani, Dell
Hongjun Zhai, ZTE
Expires: September 4, 2015 — March 5, 2015

Flexible Multilevel TRILL
(Transparent Interconnection of Lots of Links)
<draft-perlman-trill-rbridge-multilevel-09.txt>

Abstract

Extending TRILL to multiple levels has challenges that are not addressed by the already-existing capability of IS-IS to have multiple levels. One issue is with the handling of multi-destination packet distribution trees. Another issue is with TRILL switch nicknames. There have been two proposed approaches. One approach, which we refer to as the "unique nickname" approach, gives unique nicknames to all the TRILL switches in the multilevel campus, either by having the level-1/level-2 border TRILL switches advertise which nicknames are not available for assignment in the area, or by partitioning the 16-bit nickname into an "area" field and a "nickname inside the area" field. The other approach, which we refer to as the "aggregated nickname" approach, involves hiding the nicknames within areas, allowing nicknames to be reused in different areas, by having the border TRILL switches rewrite the nickname fields when entering or leaving an area. Each of those approaches has advantages and disadvantages. This informational document suggests allowing a choice of approach in each area. This allows the simplicity of the unique nickname approach in installations in which there is no danger of running out of nicknames and allows the complexity of hiding the nicknames in an area to be phased into larger installations on a per-area basis.

Status of This Memo

Table of Contents

1. Introduction

The IETF TRILL (Transparent Interconnection of Lot of Links or Tunneled Routing in the Link Layer) protocol [RFC6325] [RFC7177] provides optimal pair-wise data routing without configuration, safe forwarding even during periods of temporary loops, and support for multipathing of both unicast and multicast traffic in networks with arbitrary topology and link technology, including multi-access links. TRILL accomplishes this by using IS-IS (Intermediate System to Intermediate System [IS-IS] [RFC7176]) link state routing in conjunction with a header that includes a hop count. The design supports data labels (VLANs and Fine Grained Labels [RFC7172]) and optimization of the distribution of multi-destination data based on VLANs and multicast groups. Devices that implement TRILL are called TRILL Switches or RBridges.

Familiarity with [RFC6325] and [rfc7180bis] is assumed in this document.

1.1 TRILL Scalability Issues

There are multiple issues that might limit the scalability of a TRILL-based network:

1. the routing computation load,
2. the volatility of the link state database (LSDB) creating too much control traffic,
3. the volatility of the LSDB causing the TRILL network to be in an unconverged state too much of the time,
4. the size of the LSDB,
5. the limit of the number of TRILL switches, due to the 16-bit nickname space,
6. the traffic due to upper layer protocols use of broadcast and multicast, and
7. the size of the end node learning table (the table that remembers (egress TRILL switch, label/MAC) pairs).

Extending TRILL IS-IS to be multilevel (hierarchical) helps with all but the last of these issues.

IS-IS was designed to be multilevel [IS-IS]. A network can be partitioned into "areas". Routing within an area is known as "Level 1 routing". Routing between areas is known as "Level 2 routing". The Level 2 IS-IS network consists of Level 2 routers and links between the Level 2 routers. Level 2 routers may participate in one or more Level 1 areas, in addition to their role as Level 2 routers.

Each area is connected to Level 2 through one or more "border routers", which participate both as a router inside the area, and as a router inside the Level 2 "area". Care must be taken that it is clear, when transitioning multi-destination packets between Level 2 and a Level 1 area in either direction, that exactly one border TRILL switch will transition a particular data packet between the levels or else duplication or loss of traffic can occur.

1.2 Improvements Due to Multilevel

Partitioning the network into areas solves the first four scalability issues described above, namely,

1. the routing computation load,

2. the volatility of the LSDB creating too much control traffic,

3. the volatility of the LSDB causing the TRILL network to be in an unconverged state too much of the time,

4. the size of the LSDB.

Problem #6 in Section 1.1, namely, the traffic due to upper layer protocols use of broadcast and multicast, can be addressed by introducing a locally-scoped multi-destination delivery, limited to an area or a single link. See further discussion in Section 4.2.

Problem #5 in Section 1.1, namely, the limit of the number of TRILL switches, due to the 16-bit nickname space, will only be addressed with the aggregated nickname approach. Since the aggregated nickname approach requires some complexity in the border TRILL switches (for rewriting the nicknames in the TRILL header), the design in this document allows a campus with a mixture of unique-nickname areas, and aggregated-nickname areas. Nicknames must be unique across all Level 2 and unique-nickname area TRILL switches, whereas nicknames inside an aggregated-nickname area are visible only inside the area. Nicknames inside an aggregated-nickname area must not conflict with nicknames visible in Level 2 (which includes all nicknames inside unique nickname areas), but the nicknames inside an aggregated-nickname area may be the same as nicknames used within other aggregated-nickname areas.

TRILL switches within an area need not be aware of whether they are in an aggregated nickname area or a unique nickname area. The border TRILL switches in area A1 will claim, in their LSP inside area A1, which nicknames (or nickname ranges) are not available for choosing as nicknames by area A1 TRILL switches.

1.3 Unique and Aggregated Nickanmes

We describe two alternatives for hierarchical or multilevel TRILL. One we call the "unique nickname" alternative. The other we call the "aggregated nickname" alternative. In the aggregated nickname alternative, border TRILL switches replace either the ingress or egress nickname field in the TRILL header of unicast packets with an aggregated nickname representing an entire area.

The unique nickname alternative has the advantage that border TRILL switches are simpler and do not need to do TRILL Header nickname modification. It also simplifies testing and maintenance operations that originate in one area and terminate in a different area.

The aggregated nickname alternative has the following advantages:

- o it solves problem #5 above, the 16-bit nickname limit, in a simple way,
- o it lessens the amount of inter-area routing information that must be passed in IS-IS, and
- o it logically reduces the RPF (Reverse Path Forwarding) Check information (since only the area nickname needs to appear, rather than all the ingress TRILL switches in that area).

In both cases, it is possible and advantageous to compute multi-destination data packet distribution trees such that the portion computed within a given area is rooted within that area.

1.3 More on Areas

Each area is configured with an "area address", which is advertised in IS-IS messages, so as to avoid accidentally interconnecting areas. Although the area address had other purposes in CLNP (IS-IS was originally designed for CLNP/DECnet), for TRILL the only purpose of the area address would be to avoid accidentally interconnecting areas.

Currently, the TRILL specification says that the area address must be zero. If we change the specification so that the area address value of zero is just a default, then most of IS-IS multilevel machinery works as originally designed. However, there are TRILL-specific issues, which we address below in this document.

1.4 Terminology and Acronyms

This document generally uses the acronyms defined in [RFC6325] plus the additional acronym DBRB. However, for ease of reference, most acronyms used are listed here:

CLNP - ConnectionLess Network Protocol

DECnet - a proprietary routing protocol that was used by Digital Equipment Corporation. "DECnet Phase 5" was the origin of IS-IS.

Data Label - VLAN or Fine Grained Label [RFC7172]

DBRB - Designated Border RBridge

IS-IS - Intermediate System to Intermediate System [IS-IS]

LSDB - Link State Data Base

LSP - Link Stat PDU

PDU - Protocol Data Unit

RBridge - Routing Bridge, an alterntive name for a TRILL switch

RPF - Reverse Path Forwarding

TRILL - Transparent Interconnection of Lots of Links or Tunneled Routing in the Link Layer [RFC6325]

TRILL switch - an alternative name for an RBridge

VLAN - Virtual Local Area Network

2. Multilevel TRILL Issues

The TRILL-specific issues introduced by multilevel include the following:

a. Configuration of non-zero area addresses, encoding them in IS-IS PDUs, and possibly interworking with old TRILL switches that do not understand nonzero area addresses.

   See Section 2.1.

b. Nickname management.

   See Sections 2.5 and 2.2.

c. Advertisement of pruning information (Data Label reachability, IP multicast addresses) across areas.

   Distribution tree pruning information is only an optimization, as long as multi-destination packets are not prematurely pruned. For instance, border TRILL switches could advertise they can reach all possible Data Labels, and have an IP multicast router attached. This would cause all multi-destination traffic to be transmitted to border TRILL switches, and possibly pruned there, when the traffic could have been pruned earlier based on Data Label or multicast group if border TRILL switches advertised more detailed Data Label and/or multicast listener and multicast router attachment information.

d. Computation of distribution trees across areas for multi-destination data.

   See Section 2.3.

e. Computation of RPF information for those distribution trees.

   See Section 2.4.

f. Computation of pruning information across areas.

   See Sections 2.3 and 2.6.

g. Compatibility, as much as practical, with existing, unmodified TRILL switches.

   The most important form of compatibility is with existing TRILL fast path hardware. Changes that require upgrade to the slow path firmware/software are more tolerable. Compatibility for the relatively small number of border TRILL switches is less important than compatibility for non-border TRILL switches.

See Section 5.

2.1 Non-zero Area Addresses

The current TRILL base protocol specification [RFC6325] [RFC7177] [rfc7180bis] says that the area address in IS-IS must be zero. The purpose of the area address is to ensure that different areas are not accidentally merged. Furthermore, zero is an invalid area address for layer 3 IS-IS, so it was chosen as an additional safety mechanism to ensure that layer 3 IS-IS would not be confused with TRILL IS-IS. However, TRILL uses other techniques to avoid such confusion, such as different multicast addresses and Ethertypes on Ethernet [RFC6325], different PPP codepoints on PPP [RFC6361], and the the like, so use in TRILL of an area address that might be used in layer 3 IS-IS is not a problem.

Since current TRILL switches will reject any IS-IS messages with nonzero area addresses, the choices are as follows:

- a.1 upgrade all TRILL switches that are to interoperate in a potentially multilevel environment to understand non-zero area addresses,
- a.2 neighbors of old TRILL switches must remove the area address from IS-IS messages when talking to an old TRILL switch (which might break IS-IS security and/or cause inadvertent merging of areas),
- a.3 ignore the problem of accidentally merging areas entirely, or
- a.4 keep the fixed "area address" field as 0 in TRILL, and add a new, optional TLV for "area name" that, if present, could be compared, by new TRILL switches, to prevent accidental area merging.

In principal, different solutions could be used in different areas but it would be much simpler to adopt one of these choices uniformly.

2.2 Aggregated versus Unique Nicknames

In the unique nickname alternative, all nicknames across the campus must be unique. In the aggregated nickname alternative, TRILL switch nicknames within an aggregated area are only of local significance, and the only nickname externally (outside that area) visible is the "area nickname" (or nicknames), which aggregates all the internal nicknames.

The unique nickname approach simplifies border TRILL switches.

The aggregated nickname approach eliminates the potential problem of nickname exhaustion, minimizes the amount of nickname information that would need to be forwarded between areas, minimizes the size of the forwarding table, and simplifies RPF calculation and RPF information.

2.2.1 More Details on Unique Nicknames

With unique cross-area nicknames, it would be intractable to have a flat nickname space with TRILL switches in different areas contending for the same nicknames. Instead, each area would need to be configured with a block of nicknames. Either some TRILL switches would need to announce that all the nicknames other than that block are taken (to prevent the TRILL switches inside the area from choosing nicknames outside the area’s nickname block), or a new TLV would be needed to announce the allowable nicknames, and all TRILL switches in the area would need to understand that new TLV. An example of the second approach is given in [NickFlags].

Currently the encoding of nickname information in TLVs is by listing of individual nicknames; this would make it painful for a border TRILL switch to announce into an area that it is holding all other nicknames to limit the nicknames available within that area. The information could be encoded as ranges of nicknames to make this somewhat manageable [NickFlags]; however, a new TLV for announcing nickname ranges would not be intelligible to old TRILL switches.

There is also an issue with the unique nicknames approach in building distribution trees, as follows:

> With unique nicknames in the TRILL campus and TRILL header nicknames not rewritten by the border TRILL switches, there would have to be globally known nicknames for the trees. Suppose there are k trees. For all of the trees with nicknames located outside an area, the local trees would be rooted at a border TRILL switch or switches. Therefore, there would be either no splitting of multi-destination traffic with the area or restricted splitting of multi-destination traffic between trees rooted at a highly restricted set of TRILL switches.
>
> As an alternative, just the "egress nickname" field of multi-destination TRILL Data packets could be mapped at the border, leaving known unicast packets un-mapped. However, this surrenders much of the unique nickname advantage of simpler border TRILL switches.

Scaling to a very large campus with unique nicknames might exhaust the 16-bit TRILL nicknames space. One method might be to expand nicknames to 24bits; however, that technique would require TRILL message format changes and that all TRILL switches in the campus understand larger nicknames.

For an example of a more specific multilevel proposal using unique nicknames, see [DraftUnique].

2.2.2 More Details on Aggregated Nicknames

The aggregated nickname approach enables passing far less nickname information. It works as follows, assuming both the source and destination areas are using aggregated nicknames:

> Each area would be assigned a 16-bit nickname. This would not be the nickname of any actual TRILL switch. Instead, it would be the nickname of the area itself. Border TRILL switches would know the area nickname for their own area(s).

The TRILL Header nickname fields in TRILL Data packets being transported through a multilevel TRILL campus with aggregated nicknames are as follows:

- When both the ingress and egress TRILL switches are in the same area, there need be no change from the existing base TRILL protocol standard in the TRILL Header nickname fields.

- When being transported in Level 2, the ingress nickname is the nickname of the ingress TRILL switch's area while the egress nickname is either the nickname of the egress TRILL switch's area or a tree nickname.

- When being transported from Level 1 to Level 2, the ingress nickname is the nickname of the ingress TRILL switch itself while the egress nickname is either the nickname of the area of the egress TRILL switch or a tree nickname.

- When being transported from Level 2 to Level 1, the ingress nickname is the nickname of the ingress TRILL switch's area while the egress nickname is either the nickname of the egress TRILL switch itself or a tree nickname.

There are two variations of the aggregated nickname approach. The first is the Border Learning approach, which is described in Section 2.2.2.1. The second is the Swap Nickname Field approach, which is described in Section 2.2.2.2. Section 2.2.2.3 compares the advantages and disadvantages of these two variations of the aggregated nickname approach.

2.2.2.1 Border Learning Aggregated Nicknames

This section provides an illustrative example and description of the border learning variation of aggregated nicknames.

In the following picture, RB2 and RB3 are area border TRILL switches (RBridges). A source S is attached to RB1. The two areas have nicknames 15961 and 15918, respectively. RB1 has a nickname, say 27, and RB4 has a nickname, say 44 (and in fact, they could even have the same nickname, since the TRILL switch nickname will not be visible outside these aggreated areas).

```
        Area 15961              level 2            Area 15918
 +-------------------+     +-----------------+     +--------------+
 |                   |     |                 |     |              |
 |  S--RB1---Rx--Rz----RB2---Rb---Rc--Rd---Re--RB3---Rk--RB4---D  |
 |     27            |     |                 |     |     44       |
 |                   |     |                 |     |              |
 +-------------------+     +-----------------+     +--------------+
```

Let’s say that S transmits a frame to destination D, which is connected to RB4, and let’s say that D’s location has already been learned by the relevant TRILL switches. These relevant switches have learned the following:

1) RB1 has learned that D is connected to nickname 15918
2) RB3 has learned that D is attached to nickname 44.

The following sequence of events will occur:

- S transmits an Ethernet frame with source MAC = S and destination MAC = D.

- RB1 encapsulates with a TRILL header with ingress RBridge = 27, and egress = 15918 producing a TRILL Data packet.

- RB2 has announced in the Level 1 IS-IS instance in area 15961, that it is attached to all the area nicknames, including 15918. Therefore, IS-IS routes the packet to RB2. Alternatively, if a distinguished range of nicknames is used for Level 2, Level 1 TRILL switches seeing such an egress nickname will know to route to the nearest border router, which can be indicated by the IS-IS attached bit.

- RB2, when transitioning the packet from Level 1 to Level 2, replaces the ingress TRILL switch nickname with the area nickname, so replaces 27 with 15961. Within Level 2, the ingress RBridge field in the TRILL header will therefore be 15961, and the egress RBridge field will be 15918. Also RB2 learns that S is attached to nickname 27 in area 15961 to accommodate return traffic.

- The packet is forwarded through Level 2, to RB3, which has advertised, in Level 2, reachability to the nickname 15918.

- RB3, when forwarding into area 15918, replaces the egress nickname in the TRILL header with RB4's nickname (44). So, within the destination area, the ingress nickname will be 15961 and the egress nickname will be 44.

- RB4, when decapsulating, learns that S is attached to nickname 15961, which is the area nickname of the ingress.

Now suppose that D's location has not been learned by RB1 and/or RB3. What will happen, as it would in TRILL today, is that RB1 will forward the packet as multi-destination, choosing a tree. As the multi-destination packet transitions into Level 2, RB2 replaces the ingress nickname with the area nickname. If RB1 does not know the location of D, the packet must be flooded, subject to possible pruning, in Level 2 and, subject to possible pruning, from Level 2 into every Level 1 area that it reaches on the Level 2 distribution tree.

Now suppose that RB1 has learned the location of D (attached to nickname 15918), but RB3 does not know where D is. In that case, RB3 must turn the packet into a multi-destination packet within area 15918. In this case, care must be taken so that, in case RB3 is not the Designated transitioner between Level 2 and its area for that multi-destination packet, but was on the unicast path, that another border TRILL switch in that area not forward the now multi-destination packet back into Level 2. Therefore, it would be desirable to have a marking, somehow, that indicates the scope of this packet's distribution to be "only this area" (see also Section 4).

In cases where there are multiple transitioners for unicast packets, the border learning mode of operation requires that the address learning between them be shared by some protocol such as running ESADI [RFC7357] for all Data Labels of interest to avoid excessive unknown unicast flooding.

The potential issue described at the end of Section 2.2.1 with trees in the unique nickname alternative is eliminated with aggregated nicknames. With aggregated nicknames, each border TRILL switch that will transition multi-destination packets can have a mapping between Level 2 tree nicknames and Level 1 tree nicknames. There need not even be agreement about the total number of trees; just that the border TRILL switch have some mapping, and replace the egress TRILL switch nickname (the tree name) when transitioning levels.

2.2.2.2 Swap Nickname Field Aggregated Nicknames

As a variant, two additional fields could exist in TRILL Data packets we call the "ingress swap nickname field" and the "egress swap nickname field". The changes in the example above would be as follows:

- RB1 will have learned the area nickname of D and the TRILL switch nickname of RB4 to which D is attached. In encapsulating a frame to D, it puts the area nickname of D (15918) in the egress nickname field of the TRILL Header and puts the nickname of RB3 (44) in a egress swap nickname field.

- RB2 moves the ingress nickname to the ingress swap nickname field and inserts 15961, the area nickname for S, into the ingress nickname field.

- RB3 swaps the egress nickname and the egress swap nickname fields, which sets the egress nickname to 44.

- RB4 learns the correspondence between the source MAC/VLAN of S and the { ingress nickname, ingress swap nickname field } pair as it decapsulates and egresses the frame.

See [DraftAggregated] for a multilevel proposal using aggregated swap nicknames.

2.2.2.3 Comparison

The Border Learning variant described in Section 2.2.2.1 above minimizes the change in non-border TRILL switches but imposes the burden on border TRILL switches of learning and doing lookups in all the end station MAC addresses within their area(s) that are used for communication outside the area. This burden could be reduced by decreasing the area size and increasing the number of areas.

The Swap Nickname Field variant described in Section 2.2.2.2 eliminates the extra address learning burden on border TRILL switches but requires more extensive changes to non-border TRILL switches. In particular they must learn to associate both a TRILL switch nickname and an area nickname with end station MAC/label pairs (except for addresses that are local to their area).

The Swap Nickname Field alternative is more scalable but less backward compatible for non-border TRILL switches. It would be possible for border and other level 2 TRILL switches to support both Border Learning, for support of legacy Level 1 TRILL switches, and Swap Nickname, to support Level 1 TRILL switches that understood the

Swap Nickname method.

2.3 Building Multi-Area Trees

It is easy to build a multi-area tree by building a tree in each area separately, (including the Level 2 "area"), and then having only a single border TRILL switch, say RBx, in each area, attach to the Level 2 area. RBx would forward all multi-destination packets between that area and Level 2.

People might find this unacceptable, however, because of the desire to path split (not always sending all multi-destination traffic through the same border TRILL switch).

This is the same issue as with multiple ingress TRILL switches injecting traffic from a pseudonode, and can be solved with the mechanism that was adopted for that purpose: the affinity TLV [DraftCMT]. For each tree in the area, at most one border RB announces itself in an affinity TLV with that tree name.

2.4 The RPF Check for Trees

For multi-destination data originating locally in RBx's area, computation of the RPF check is done as today. For multi-destination packets originating outside RB1's area, computation of the RPF check must be done based on which one of the border TRILL switches (say RB1, RB2, or RB3) injected the packet into the area.

A TRILL switch, say RB4, located inside an area, must be able to know which of RB1, RB2, or RB3 transitioned the packet into the area from Level 2. (or into Level 2 from an area).

This could be done based on having the DBRB announce the transitioner assignments to all the TRILL switches in the area, or the Affinity TLV mechanism given in [DraftCMT], or the New Tree Encoding mechanism discussed in Section 4.1.1.

2.5 Area Nickname Acquisition

In the aggregated nickname alternative, each area must acquire a unique area nickname. It is probably simpler to allocate a block of nicknames (say, the top 4000) to be area addresses, and not used by any TRILL switches.

The area nicknames need to be advertised and acquired through Level 2.

Within an area, all the border TRILL switches must discover each other through the Level 1 link state database, by using the IS-IS attach bit or by explicitly advertising in their LSP "I am a border RBridge".

Of the border TRILL switches, one will have highest priority (say RB7). RB7 can dynamically participate, in Level 2, to acquire a pseudo-nickname for the area analagous to the pseudo-nickname for an active-active edge group [PseudoNickname]. Alternatively, RB7 could give the area a pseudonode IS-IS ID, such as RB7.5, within Level 2. So an area would appear, in Level 2, as a pseudonode and the pseudonode can participate, in Level 2, to acquire a nickname for the area.

Within Level 2, all the border TRILL switches for an area can advertise reachability to the area, which would mean connectivity to the area nickname.

2.6 Link State Representation of Areas

Within an area, say area A1, there is an election for the DBRB, (Designated Border RBridge), say RB1. This can be done through LSPs within area A1. The border TRILL switches announce themselves, together with their DBRB priority. (Note that the election of the DBRB cannot be done based on Hello messages, because the border TRILL switches are not necessarily physical neighbors of each other. They can, however, reach each other through connectivity within the area, which is why it will work to find each other through Level 1 LSPs.)

RB1 acquires the area nickname (in the aggregated nickname approach) and may give the area a pseudonode IS-IS ID (just like the DRB would give a pseudonode IS-IS ID to a link) depending on how the area nickname is handled. RB1 advertises, in area A1, the area nickname that RB1 has acquired (and what the pseudonode IS-IS ID for the area is if needed).

Level 1 LSPs (possibly pseudonode) initiated by RB1 for the area include any information external to area A1 that should be input into area A1 (such as area nicknames of external areas, or perhaps (in the unique nickname variant) all the nicknames of external TRILL switches in the TRILL campus and pruning information such as multicast listeners and labels). All the other border TRILL switches for the area announce (in their LSP) attachment to that area.

Within Level 2, RB1 generates a Level 2 LSP on behalf of the area.

The same pseudonode ID could be used within Level 1 and Level 2, for the area. (There does not seem any reason why it would be useful for it to be different, but there's also no reason why it would need to be the same). Likewise, all the area A1 border TRILL switches would announce, in their Level 2 LSPs, connection to the area.

3. Area Partition

It is possible for an area to become partitioned, so that there is still a path from one section of the area to the other, but that path is via the Level 2 area.

With multilevel TRILL, an area will naturally break into two areas in this case.

Area addresses might be configured to ensure two areas are not inadvertently connected. Area addresses appears in Hellos and LSPs within the area. If two chunks, connected only via Level 2, were configured with the same area address, this would not cause any problems. (They would just operate as separate Level 1 areas.)

A more serious problem occurs if the Level 2 area is partitioned in such a way that it could be healed by using a path through a Level 1 area. TRILL will not attempt to solve this problem. Within the Level 1 area, a single border RBridge will be the DBRB, and will be in charge of deciding which (single) RBridge will transition any particular multi-destination packets between that area and Level 2. If the Level 2 area is partitioned, this will result in multi-destination data only reaching the portion of the TRILL campus reachable through the partition attached to the TRILL switch that transitions that packet. It will not cause a loop.

4. Multi-Destination Scope

There are at least two reasons it would be desirable to be able to mark a multi-destination packet with a scope that indicates the packet should not exit the area, as follows:

1. To address an issue in the border learning variant of the aggregated nickname alternative, when a unicast packet turns into a multi-destination packet when transitioning from Level 2 to Level 1, as discussed in Section 4.1.

2. To constrain the broadcast domain for certain discovery, directory, or service protocols as discussed in Section 4.2.

Multi-destination packet distribution scope restriction could be done in a number of ways. For example, there could be a flag in the packet that means "for this area only". However, the technique that might require the least change to TRILL switch fast path logic would be to indicate this in the egress nickname that designates the distribution tree being used. There could be two general tree nicknames for each tree, one being for distribution restricted to the area and the other being for multi-area trees. Or there would be a set of N (perhaps 16) special currently reserved nicknames used to specify the N highest priority trees but with the variation that if the special nickname is used for the tree, the packet is not transitioned between areas. Or one or more special trees could be built that were restricted to the local area.

4.1 Unicast to Multi-destination Conversions

In the border learning variant of the aggregated nickname alternative, a unicast packet might be known at the Level 1 to Level 2 transition, be forwarded as a unicast packet to the least cost border TRILL switch advertising connectivity to the destination area, but turn out to have an unknown destination { MAC, Data Label } pair when it arrives at that border TRILL switch.

In this case, the packet must be converted into a multi-destination packet and flooded in the destination area. However, if the border TRILL switch doing the conversion is not the border TRILL switch designated to transition the resulting multi-destination packet, there is the danger that the designated transitioner may pick up the packet and flood it back into Level 2 from which it may be flooded into multiple areas. This danger can be avoided by restricting any multi-destination packet that results from such a conversion to the destination area through a flag in the packet or though distributing it on a tree that is restricted to the area, or other techniques (see Section 4).

Alternatively, a multi-destination packet intended only for the area could be tunneled (within the area) to the RBridge RBx, that is the appointed transitioner for that form of packet (say, based on VLAN or FGL), with instructions that RBx only transmit the packet within the area, and RBx could initiate the multi-destination packet within the area. Since RBx introduced the packet, and is the only one allowed to transition that packet to Level 2, this would accomplish scoping of the packet to within the area. Since this case only occurs in the unusual case when unicast packets need to be turned into multi-destination as described above, the suboptimality of tunneling between the border TRILL switch that receives the unicast packet and the appointed level transitioner for that packet, would not be an issue.

4.1.1 New Tree Encoding

The current encoding, in a TRILL header, of a tree, is of the nickname of the tree root. This requires all 16 bits of the egress nickname field. TRILL could instead, for example, use the bottom 6 bits to encode the tree number (allowing 64 trees), leavinig 10 bits to encode information such as:

- o scope: a flag indicating whether it should be single area only, or entire campus
- o border injector: an indicator of which of the k border TRILL switches injected this packet

If TRILL were to adopt this new encoding, it would also avoid the limitations of the Affinity sub-TLV [DraftCMT] in the single area case [PseudoNickname]; any of the TRILL switches in an edge group could inject a multi-destination packet. This would require all TRILL switches to be changed to understand the new encoding for a tree, and it would require a TLV in the LSP to indicate which number each of the TRILL switches in an edge group would be.

4.2 Selective Broadcast Domain Reduction

There are a number of service, discovery, and directory protocols that, for convenience, are accessed via multicast or broadcast frames. Examples are DHCP, the NetBIOS Service Location Protocol, and multicast DNS.

Some such protocols provide means to restrict distribution to an IP subnet or equivalent to reduce size of the broadcast domain they are using and then provide a proxy that can be placed in that subnet to use unicast to access a service elsewhere. In cases where a proxy mechanism is not currently defined, it may be possible to create one that references a central server or cache. With multilevel TRILL, it is possible to construct very large IP subnets that could become saturated with multi-destination traffic of this type unless packets can be further restricted in their distribution. Such restricted distribution can be accomplished for some protocols, say protocol P, in a variety of waying including the following:

- Either (1) at all ingress TRILL switches in an area place all protocol P multi-destination packets on a distribution tree in such a way that the packets are restricted to the area or (2) at all border TRILL switches between that area and Level 2, detect protocol P multi-destination packets and do not transition them.

- Then place one, or a few for redundancy, protocol P proxyies inside each area where protocol P may be in use. These proxies unicast protocol P requests or other messages to the actual campus server(s) for P. They also receive unicast responses or other messages from those servers and deliver them within the area via unicast, multicast, or broadcast as appropriate. (Such proxies would not be needed if it was acceptable for all protocol P traffic to be restricted to an area.)

While it might seem logical to connect the campus servers to TRILL switches in Level 2, they could be placed within one or more areas so that, in some cases, those areas might not require a local proxy server.

5. Co-Existence with Old TRILL switches

TRILL switches that are not multilevel aware may have a problem with calculating RPF Check and filtering information, since they would not be aware of assignment of border TRILL switch transitioning.

A possible solution, as long as any old TRILL switches exist within an area, is to have the border TRILL switches elect a single DBRB (Designated Border RBridge), and have all inter-area traffic go through the DBRB (unicast as well as multi-destination). If that DBRB goes down, a new one will be elected, but at any one time, all inter-area traffic (unicast as well as multi-destination) would go through that one DRBR. However this eliminates load splitting at level transition.

6. Multi-Access Links with End Stations

Care must be taken, in the case where there are multiple TRILL switches on a link with end stations, that only one TRILL switch ingress/egress any given data packet from/to the end nodes. With existing, single level TRILL, this is done by electing a single Designated RBridge per link, which appoints a single Appointed Forwarder per VLAN [RFC7177] [RFC6439]. But suppose there are two (or more) TRILL switches on a link in different areas, say RB1 in area 1000 and RB2 in area 2000, and that the link contains end nodes. If RB1 and RB2 ignore each other's Hellos then they will both ingress/egress end node traffic from the link.

A simple rule is to use the TRILL switch or switches having the lowest numbered area, comparing area numbers as unsigned integers, to handle native traffic. This would automatically give multilevel-ignorant legacy TRILL switches, that would be using area number zero, highest priority for handling end stations, which they would try to do anyway.

Other methods are possible. For example doing the selection of Appointed Forwarders and of the TRILL switch in charge of that selection across all TRILL switches on the link regardless of area. However, a special case would then have to be made in any case for legacy TRILL switches using area number zero.

Any of these techniques require multilevel aware RBridges to take actions based on Hellos from from RBridges in other areas even though they will not form an adjacency with such RBridges.

7. Summary

This draft discusses issues and possible approaches to multilevel TRILL. The alternative using aggregated areas has significant advantages in terms of scalability over using campus wide unique nicknames, not just of avoiding nickname exhaustion, but by allowing RPF Checks to be aggregated based on an entire area; however, the alternative using unique nicknames is simpler and avoids the changes in border TRILL switches required to support aggregated nicknames. It is possible to support both. For example, a TRILL campus could use simpler unique nicknames until scaling begins to cause problems and then start to introduce areas with aggregated nicknames.

Some issues are not difficult, such as dealing with partitioned areas. Some issues are more difficult, especially dealing with old TRILL switches.

8. Security Considerations

This informational document explores alternatives for the use of multilevel IS-IS in TRILL. It does not consider security issues. For general TRILL Security Considerations, see [RFC6325].

9. IANA Considerations

This document requires no IANA actions.

Normative References

[IS-IS] - ISO/IEC 10589:2002, Second Edition, "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)", 2002.

[RFC6325] - Perlman, R., Eastlake 3rd, D., Dutt, D., Gai, S., and A. Ghanwani, "Routing Bridges (RBridges): Base Protocol Specification", RFC 6325, July 2011.

[RFC6439] - Perlman, R., Eastlake, D., Li, Y., Banerjee, A., and F. Hu, "Routing Bridges (RBridges): Appointed Forwarders", RFC 6439, November 2011.

[rfc7180bis] - D. Eastlake, M. Zhang, et al, "TRILL: Clarifications, Corrections, and Updates", draft-ietf-trill-rfc7180bis, work in progress

Informative References

[RFC6361] - Carlson, J. and D. Eastlake 3rd, "PPP Transparent Interconnection of Lots of Links (TRILL) Protocol Control Protocol", RFC 6361, August 2011.

[RFC7172] - Eastlake 3rd, D., Zhang, M., Agarwal, P., Perlman, R., and D. Dutt, "Transparent Interconnection of Lots of Links (TRILL): Fine-Grained Labeling", RFC 7172, May 2014

[RFC7176] - Eastlake 3rd, D., Senevirathne, T., Ghanwani, A., Dutt, D., and A. Banerjee, "Transparent Interconnection of Lots of Links (TRILL) Use of IS-IS", RFC 7176, May 2014.

[RFC7177] - Eastlake 3rd, D., Perlman, R., Ghanwani, A., Yang, H., and V. Manral, "Transparent Interconnection of Lots of Links (TRILL): Adjacency", RFC 7177, May 2014, <http://www.rfc-editor.org/info/rfc7177>.

[RFC7357] - Zhai, H., Hu, F., Perlman, R., Eastlake 3rd, D., and O. Stokes, "Transparent Interconnection of Lots of Links (TRILL): End Station Address Distribution Information (ESADI) Protocol", RFC 7357, September 2014, <http://www.rfc-editor.org/info/rfc7357>.

[DraftAggregated] - Bhargav Bhikkaji, Balaji Venkat Venkataswami, Narayana Perumal Swamy, "Connecting Disparate Data Center/PBB/Campus TRILL sites using BGP", draft-balaji-trill-over-ip-multi-level, Work In Progress.

[DraftCMT] - Tissa Senevirathne, Janardhanan Pathang, Jon Hudson, "Coordinated Multicast Trees (CMT) for TRILL", draft-tissa-trill-cmt, Work in Progress.

[DraftUnique] - Tissa Senevirathne, Les Ginsberg, Janardhanan Pathangi, Jon Hudson, Sam Aldrin, Ayan Banerjee, Sameer Merchant, "Default Nickname Based Approach for Multilevel TRILL", draft-tissa-trill-multilevel, Work In Progress.

[PseudoNickname] - H. Zhai, T. Senevirathne, et al, "TRILL: Pseudo-Nickname for Active-active Access", draft-ietf-trill-pseudonode-nickname, work in progress.

[NickFlags] - Eastlake, D., W. Hao, draft-eastlake-trill-nick-label-prop, Work In Progress.

Acknowledgements

The helpful comments of the following are hereby acknowledged: David Michael Bond and Dino Farinacci.

The document was prepared in raw nroff. All macros used were defined within the source file.

Authors' Addresses

Radia Perlman
EMC
2010 256th Avenue NE, #200
Bellevue, WA 98007 USA

EMail: radia@alum.mit.edu

Donald Eastlake
Huawei Technologies
155 Beaver Street
Milford, MA 01757 USA

Phone: +1-508-333-2270
Email: d3e3e3@gmail.com

Mingui Zhang
Huawei Technologies
No.156 Beiqing Rd. Haidian District,
Beijing 100095 P.R. China

EMail: zhangmingui@huawei.com

Anoop Ghanwani
Dell
5450 Great America Parkway
Santa Clara, CA 95054 USA

EMail: anoop@alumni.duke.edu

Hongjun Zhai
ZTE
68 Zijinghua Road, Yuhuatai District
Nanjing, Jiangsu 210012 China

Phone: +86 25 52877345
Email: zhai.hongjun@zte.com.cn

Copyright and IPR Provisions

TRILL Working Group   Yizhou Li
INTERNET-DRAFT   Donald Eastlake
Intended Status: Standard Track   Linda Dunbar
                                                   Huawei Technologies
                                                         Radia Perlman
                                                                   EMC
                                                       Igor Gashinsky
                                                                 Yahoo
Expires: August 19, 2015   February 15, 2015

TRILL: ARP/ND Optimization
draft-yizhou-trill-arp-optimization-01

Abstract

This document describes mechanisms to optimize the ARP (Address Resolution Protocol) and ND (Neighbor Discovery) traffic in TRILL campus. Such optimization reduces packet flooding over a TRILL campus.

Table of Contents

1 Introduction

ARP [RFC826] and ND [RFC4861] are normally sent by broadcast and multicast respectively. To reduce the burden on a TRILL campus caused by these multi-destination messages, RBridges MAY implement an "optimized ARP/ND response", as specified herein, when the target's location is known by the ingress RBridge or can be obtained from a directory. This avoids ARP/ND query flooding.

1.1 Terminology

The acronyms and terminology in [RFC6325] is are used herein. Some of these are listed below for convenience with the following along with some additions:

Campus: a TRILL network consisting of TRILL switches, links, and possibly bridges bounded by end stations and IP routers. For TRILL, there is no "academic" implication in the name "campus".

Data Label - VLAN or FGL.

ARP - Address Resolution Protocol [RFC826].

ESADI - End Station Address Distribution Information [RFC7357].

FGL - Fine-Grained Label [RFC7172].

IA - Interface Addresses, a TRILL APPsub-TLV [IA].

ND - Neighbor Discoery [RFC4861].

RBridge - Routing Bridge, an alternative term for a TRILL switch.

TRILL - Transparent Interconnection of Lots of Links or Tunneled Routing in the Link Layer.

TRILL switch -- a device implementing the TRILL protocol, an alternative term for an RBridge.

2 IP/MAC Address Mappings

Traditionally an RBridge learns the MAC and and Data Label (VLAN or FGL) to nickname correspondence of a remote host, as per [RFC6325] and [RFC7172], from TRILL data frames received. No IP address information is learned directly from the TRILL data frame. Interface

Addresses (IA) APPsub-TLV [IA] enhances the TRILL base protocol by allowing IP and MAC address mappings to be distributed in the control plane by any RBridge. This APPsub-TLV appears inside the TRILL GENINFO TLV in ESADI [RFC7357] but the value data structure it specifies may also occur in other application contexts. Edge Directory Assist Mechanisms [DirMech] makes use of this APPsub-TLV for its push model and uses the value data structure it specifies in its pull model.

An RBridge can easily know the IP/MAC address mappings of the local hosts that it is attached to it via its access ports by receiving ARP [RFC826] or ND [RFC4861] messages. If the RBridge has extracted the sender's IP/MAC address pair from the received data packet, it may save the information and use the IA APPsub-TLV to distribute it to other RBridges through ESADI. Then the relevant remote RBridges (normally those interested in the same Data Label as the original ARP/ND messages) receive and save such mapping information also. There are others ways that RBridges save IP/MAC address mappings in advance, e.g. import from management system and distribution by directory servers [DirMech].

The examples given above shows that RBridges may have saved a host's triplet of {IP address, MAC address, ingress nickname} for a given Data Label (VLAN or FGL) before that host sends or receives any real data packet. Note such information may or may not be a complete list and may or may not exist on all RBridges. The information may be possibly from different sources. RBridges can then use the Flags Field in IA APPsub-TLV to identify if the source is a directory server or local observation by the sender. Different confidence level may also be used to indicate the reliability of the mapping information.

3 Handling ARP/ND Messages

A native frame that is an ARP [RFC826] message is detected by its Ethertype of 0x0806. A native frame that is an ND [RFC4861] is detected by being one of five different ICMPv6 packet types. ARP/ND is commonly used on a link to (1) query for the MAC address corresponding to an IPv4 or IPv6 address, (2) test if an IPv4/IPv6 address is already in use, or (3) to announce the new or updated info on any of IPv4/IPv6 address, MAC address, and/or point of attachment.

To simplify the text, we use the following terms in this section.

1) IP address - indicated protocol address that is normally an IPv4 address in ARP or an IPv6 address in ND.

2) sender’s IP/MAC address - sender protocol/hardware address in ARP, source IP address and source link-layer address in ND

3) target’s IP/MAC address - target protocol/hardware address in ARP, target address and target link-layer address in ND

When an ingress RBridge receives an ARP/ND message, it can perform the steps described in the sub-sections below.

3.1 Get Sender’s IP/MAC Mapping Information for Non-zero IP

If the sender’s MAC has not been saved by the ingress RBridge before, populate the information of sender’s IP/MAC in its ARP table;

else if the sender’s MAC has been saved before but with a different IP address mapped, the RBridge should verify if a duplicate IP address has already been in use. The RBridge may use different strategies to do so, for example, ask an authoritative entity like directory servers or encapsulate and unicast the ARP/ND message to the location where it believes a duplicate address is in use.

The ingress RBridge may use the IA APPsub-TLV [IA] with the Local flag set in ESADI [RFC7357] to distribute any new or updated IP/MAC information obtained in this step. If a push directory server is used, such information can be distributed as per [DirMech].

3.2 Determine How to Reply to ARP/ND

a) If the message is a generic ARP/ND request and the ingress RBridge knows the target’s IP address, the ingress RBridge may decide to take one or a combination of the following actions:

a.1. Send an ARP/ND response directly to the querier, with the target’s MAC address, as believed by the ingress RBridge.

a.2. Encapsulate the ARP/ND request to the target’s Designated RBridge, and have the egress RBridge for the target forward the query to the target. This behavior has the advantage that a response to the request is authoritative. If the request does not reach the target, then the querier does not get a response.

a.3. Block ARP/ND requests that occur for some time after a request to the same target has been launched, and then respond to the querier when the response to the recently-launched query to that target is received.

a.4. Pull the most up-to-date records if a pull directory server is available [DirMech] and reply to the querier.

a.5. Flood the request as per [RFC6325].

b) If the message is a generic ARP request and the ingress RBridge does not know target's IP address, the ingress RBridge may take one of the following actions.

b.1. Flood the message as per [RFC6325].

b.2. Use directory server to pull the information [DirMech] and reply to the querier.

b.3. Drop the message.

c) If the message is a gratuitous ARP which can be identified by the same sender's and target's "protocol" address fields or an Unsolicited Neighbor Advertisements [RFC4861] in ND:

The RBridge may use an IA APPsub-TLV [IA] with the Local flag set to distribute the sender's MAC and IP mapping information. When one or more directory servers are deployed and complete Push Directory information is used by all the TRILL switches in the Data Label, a gratuitous ARP or unsolicited NA SHOULD be discarded rather than ingressed. Otherwise, they are either ingressed and flooded as per [RFC6325] or discarded depending on local policy.

d) If the message is a Address Probe ARP Query [RFC5227] which can be identified by the sender's protocol (IPv4) address field being zero and the target's protocol address field being the IPv4 address to be tested or a Neighbor Solicitation for DAD (Duplicate Address Detection) which has the unspecified source address [RFC4862]: it should be handled as the generic ARP message as in a) and b).

It should be noted in the case of secure neighbor discovery (SEND) [RFC3971], cryptography might prevent local reply by the ingress RBridge, since the RBridge would not be able to sign the response with the target's private key.

It is not essential that all RBridges use the same strategy for which option to select for a particular ARP/ND query. It is up to the implementation.

3.3 Determine How to Handle the ARP/ND Response

If the ingress RBridge R1 decides to unicast the ARP/ND request to the target's egress RBridge R2 as discussed in subsection 3.2 item a) or to flood the request as per [RFC6325], then R2 decapsulates the query, and initiate an ARP/ND query on the target’s link. When/if the target responds, R2 encapsulates and unicasts the response to R1, which decapsulates the response and sends it to the querier. R2 should initiates a link state update to inform all the other RBridges of the target’s location, layer 3 address, and layer 2 address, in addition to forwarding the reply to the querier. The update message can be carried by an IA APPsub-TLV [IA] with the Local flag set in ESADI [RFC7357] or as per [DirMech] if push directory server is in use.

4 Handling RARP (Reverse Address Resolution Protocol) Messages

RARP [RFC903] uses the same packet format as ARP but a different Ethertype (0x8035) and opcode values. Its use is similar to the generic ARP Request/Response as described in 3.2 a) and b). The difference is that it is intended to query for the target "protocol" address corresponding to the target "hardware" address provided. It should be handled by doing a local cache or directory server lookup on the target "hardware" address provided to find a mapping to the desired "protocol" address. Normally, it is used to look up a MAC address to find the corresponding IP address.

5 Security Considerations

ARP and ND messages can be easily forged. Therefore the learning of MAC/IP addresses from them should not be considered as reliable. RBridge can use the confidence level in IA APPsub-TLV information received via ESADI or pull directory retrievals to determine the reliability of MAC/IP address mapping. (ESADI information can be secured as provide in [RFC7357] and pull directory information can be secured as provide in [DirMech].) It is up to the implementation to decide if an RBridge should distribute the IP and MAC address mappings received from local native ARP/ND messages to other RBridges in the same Data Label.

The ingress RBridge should also rate limit the ARP/ND queries for the same target to be injected into the TRILL campus to prevent possible denial of service attacks.

The ingress RBridge should also rate limit the ARP/ND queries for the same target to be injected to the TRILL campus prevent the possible attack.

6 IANA Considerations

No IANA action is required. RFC Editor: please delete this section before publication.

7 References

7.1 Normative References

[RFC826] Plummer, D., "An Ethernet Address Resolution Protocol", RFC 826, November 1982.

[RFC903] Finlayson, R., Mann, T., Mogul, J., and M. Theimer, "A Reverse Address Resolution Protocol", STD 38, RFC 903, June 1984

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.

[RFC6165] Banerjee, A. and D. Ward, "Extensions to IS-IS for Layer-2 Systems", RFC 6165, April 2011.

[RFC6325] Perlman, R., et.al. "RBridge: Base Protocol Specification", RFC 6325, July 2011.

[RFC6439] Eastlake, D. et.al., "RBridge: Appointed Forwarder", RFC 6439, November 2011.

[RFC7172] Eastlake 3rd, D., Zhang, M., Agarwal, P., Perlman, R., and D. Dutt, "Transparent Interconnection of Lots of Links (TRILL): Fine-Grained Labeling", RFC 7172, May 2014, <http://www.rfc-editor.org/info/rfc7172>.

7.2 Informative References

[RFC3971] Arkko, J., Ed., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, March 2005.

[RFC5227] Cheshire, S., "IPv4 Address Conflict Detection", RFC 5227, July 2008.

[RFC7067] Dunbar, L., Eastlake 3rd, D., Perlman, R., and I. Gashinsky, "Directory Assistance Problem and High-Level Design Proposal", RFC 7067, November 2013.

[IA] Eastlake, D., Li Y., R. Perlman, "TRILL: Interface Addresses APPsub-TLV", draft-eastlake-trill-ia-appsubtlv, work in progress.

[DirMech] Dunbar, L., Eastlake 3rd, D., Perlman, R., I. Gashinsky. and Li Y., TRILL: Edge Directory Assist Mechanisms", draft-ietf-trill-directory-assist-mechanisms, work in progress.

Authors' Addresses

Yizhou Li
Huawei Technologies
101 Software Avenue,
Nanjing 210012
China

Phone: +86-25-56625375
EMail: liyizhou@huawei.com

Donald Eastlake
Huawei R&D USA
155 Beaver Street
Milford, MA 01757 USA

Phone: +1-508-333-2270
Email: d3e3e3@gmail.com

Linda Dunbar
Huawei Technologies
5430 Legacy Drive, Suite #175
Plano, TX 75024, USA

Phone: +1-469-277-5840
EMail: ldunbar@huawei.com

Radia Perlman
EMC
2010 256th Avenue NE, #200
Bellevue, WA 98007
USA

Email: Radia@alum.mit.edu

Igor Gashinsky
Yahoo
45 West 18th Street 6th floor
New York, NY 10011 USA

EMail: igor@yahoo-inc.com

TRILL Working Group   Y. Li
INTERNET-DRAFT   D. Eastlake
Intended Status: Standard Track   W. Hao
                                                               H. Chen
                                                   Huawei Technologies
                                                          Radia Perlman
                                                                   EMC
                                                          Naveen Nimmu
                                                              Broadcom
                                                         S. Chatterjee
                                                                 Cisco
                                                     Sunny Rajagopalan
                                                                   IBM
Expires: September 4, 2015   March 3, 2015

TRILL: Data Label based Tree Selection for Multi-destination Data
draft-yizhou-trill-tree-selection-04

Abstract

TRILL uses distribution trees to deliver multi-destination frames. Multiple trees can be used by an ingress RBridge for flows regardless of the VLAN, Fine Grained Label (FGL), and/or multicast group of the flow. Different ingress RBridges may choose different distribution trees for TRILL Data packets in the same VLAN, FGL, and/or multicast group. To avoid unnecessary link utilization, distribution trees should be pruned based on VLAN and/or FGL and/or multicast destination address. If any VLAN, FGL, or multicast group can be sent on any tree, for typical fast path hardware, the amount of pruning information is multiplied by the number of tree; however, there is a limited capacity for such pruning information.

This document specifies an optional facility to restrict the TRILL Data packets sent on particular distribution trees by VLAN, FGL, and/or multicast group thus reducing the total amount of pruning information so that it can more easily be accommodated by fast path hardware.

Status of this Memo

Copyright and License Notice

Table of Contents

1. Introduction

1.1. Background Description

One or more distribution trees, identified by their root nickname, are used to distribute multi-destination data in a TRILL campus [RFC6325]. The RBridge having the highest tree root priority announces the total number of trees that should be computed for the campus. It may also specify the ordered list of trees that RBridges need to compute using the Tree Identifiers (TREE-RT-IDs) sub-TLV [RFC7176]. Every RBridge can specify the trees it will use in the Trees Used Identifiers (TREE-USE-IDs) sub-TLV and the VLANs or fine grained labels (FGLs [RFC7172]) it is interested in are specified in Interested VLANs and/or Interested Labels sub-TLVs [RFC7176]. It is suggested that, by default, the ingress RBridge use the distribution tree whose root is the closest [RFC6325]. Trees Used Identifiers sub-TLVs are used to build the RPF Check table that is used for reverse path forwarding check; Interested VLANs and Interested Labels sub-TLVs are used for distribution tree pruning and the multi-destination forwarding table with pruning info is built based on that. Each distribution tree SHOULD be pruned per VLAN/FGL, eliminating branches that have no potential receivers downstream [RFC6325]. Further pruning based on Layer 2 or Layer 3 multicast address is also possible.

Defaults are provided but it is implementation dependent how many trees to calculate, where the tree roots are located, and which tree(s) are to be used by an ingress RBridge. With the increasing demand to use TRILL in data center networks, there are some features we can explore for multi-destination frames in the data center use case. In order to achieve non-blocking data forwarding, a fat tree structure is often used. Figure 1 shows a typical fat tree structure based data center network. RB1 and RB2 are aggregation switches and RB11 to RB14 are access switches. It is a common practice to configure the tree roots to be at the aggregation switches for more efficient traffic transportation. All the ingress RBridges that are access switches have the same distance to all the tree roots.

```
          +-----+    +-----+
          | RB1 |    | RB2 |
          +-----+    +-----+
           / | \\     / /|\
          /  |  \ \  / / | \
         /   |   \  \ /  |  \-----+
        /    |    \/  \  |        |
       /     |    /\/   \|        |
      /  /---+---/ /\    |\       |
     /  /    |    /  \   |  \     |
    /  /     |   /    \  |    \   |
   /  /      |  /      \ |      \ |
+-----+   +-----+   +-----+   +-----+
| RB11|   | RB12|   | RB13|   | RB14|
+-----+   +-----+   +-----+   +-----+
```

Figure 1. Fat Tree Structure based TRILL network

1.2. Motivations

In the structure of figure 1, if we choose to put the tree roots at RB1 and RB2, the ingress RBridge (e.g. RB11) would find more than one closest tree root (i.e. RB1 & RB2). An ingress RBridge has two options to select the tree root for multi-destination frames: choose one and only one as distribution tree root or use ECMP-like algorithm to balance the traffic among the multiple trees whose roots are at the same distance.

- For the former, a single tree used by each ingress RBridge, can have the obvious problem of inefficient link usage. For example, if RB11 chooses the tree1 that is rooted at RB1 as the distribution tree, the link between RB11 and RB2 will never be used for multi-destination frames ingressed by RB11.

- For the latter, ECMP based tree selection results in a linear increase in multicast forwarding table size with the number of trees as explained in the next paragraph.

A multicast forwarding table at an RBridge is normally used to map the key of (tree nickname + VLAN) to an index to a list of ports for multicast packet replication. The key used for mapping is simply the tree nickname when the RBridge does not prune the tree and the key could be (tree nickname + VLAN + Layer 2 or 3 multicast address) when the RBridge was programmed by control plane with Layer 2 or 3 multicast pruning information.

For any RBridge RBn, for each VLAN x, if RBn is in a distribution tree t for VLAN x, there will be an entry of (t, x, port list) in the multicast forwarding table on RBn. Typically each entry contains a distinct combination of (tree nickname, VLAN) as the lookup key. If there are n such trees and m such VLANs, the multicast forwarding table size on RBn is n*m entries. If fine-grained label is used [RFC7172] and/or finer pruning is used (for example, VLAN + multicast group address is used for pruning), the value of m increases. In the larger scale data center, more trees would be necessary for better load balancing purpose and it results in the increasing of value n. In either case, the number of table entries n*m will increase dramatically.

The left table in Figure 2 shows an example of the multicast forwarding table on RB11 in the Figure 1 topology with 2 distribution trees in a campus using typical fast path hardware. The number of entries is approximately 2 * 4K in this case. If 4 distribution trees are used in a TRILL campus and RBn has 4K VLANs with downstream receivers, it consumes 16K table entries. TRILL multicast forwarding tables have a limited size in hardware implementation. The table entries are a precious resource. In some implementations, the table is shared with Layer 3 IP multicast for a total of 16K or 8K table entries. Therefore we want to reduce the table size consumed as much as possible and at the same time maintain the load balancing among trees.

In cases where blocks of consecutive VLANs or FGLs can be assigned to a tree, it would be very helpful in compressing the multicast forwarding table if entries could have a Data Label value and mask and the fast path hardware could do longest prefix matching. But few if any fast path implementations provide such logic.

A straightforward way to alleviate the limited table entries problem is not to prune the distribution tree. However this can only be used in the restricted scenarios for the following reasons:

- Not pruning unnecessarily wastes bandwidth for multi-destination packets. There is broadcast traffic in each VLAN, like ARP and unknown unicast. In addition, if there is a lot of Layer 3 multicast traffic in some VLAN, no pruning may result in the worse consequence of Layer 3 user data unnecessarily flooded over the campus. The volume could be huge if certain applications like IPTV are supported. Finer pruning like pruning based on multicast group may be desirable in this case.

- Not pruning is only useful at pure transit nodes. Edge nodes always need to maintain the multicast forwarding table with the key of (tree nickname + VLAN) since the edge node needs to decide whether and how to replicate the frame to local access ports based on VLAN. It is very likely that edge nodes are relatively low scale switches with the smaller shared table size, say 4K, available.

- Security concerns. VLAN based traffic isolation is a basic requirement in some scenarios. No pruning may result in the unnecessary leakage of the traffic. Misbehaved RBridges may take advantage of this.

In addition to the multicast table size concern, some silicon does not currently support hashing-based tree nickname selection at the ingress RBridge. VLAN based tree selection is used instead. The control plane of the ingress RBridge maps the incoming VLAN x to a tree nickname t. Then the data plane will always use tree t for VLAN x multi-destination frames. Though an ingress RBridge may choose multiple trees to be used for load sharing, it can use one and only one tree for each VLAN. If we make sure all ingress RBridges campus-wide send VLAN x multi-destination packets only using tree t, then there would be no need to store the multicast table entry with the key of (tree-other- than-t, x) on any RBridge.

This document describes the TRILL control plane support for a VLAN based tree selection mechanism to reduce the multicast forwarding table size. It is compatible with the silicon implementation mentioned in the previous paragraph. Here VLAN based tree selection is a general term which also includes finer granularity case such as VLAN + Layer 2 or 3 multicast or FGL group based selection.

2. Terminology Used in This Document

This document uses the terminology from [RFC6325] and [RFC7172], some of which is repeated below for convenience, along with some additional terms listed below:

campus: Name for a TRILL network, like "bridged LAN" is a name for a bridged network. It does not have any academic implication.

Data Label: VLAN or FGL.

ECMP: Equal Cost Multi-Path [RFC6325].

FGL: Finge Grainge Lable [RFC7172].

IPTV: "Television" (video) over IP.

RBridge: An alternative name for a TRILL switch.

TRILL: Transparent Interconnection of Lots of Links (or Tunneled Routing in the Link Layer).

TRILL switch: A device implementing the TRILL protocol. Sometimes called an RBridge.

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 RFC-2119 [RFC2119].

3. Data Label based Tree Selection

Data Label based tree selection can be used as a complementary distribution tree selection mechanism, especially when the multicast forwarding table size is a concern.

3.1 Overview

The tree root with the highest priority announces the tree nicknames and the Data Labels allowed on each tree. Such tree to Data Label correspondence announcements can be based on static configuration or some predefined algorithm beyond the scope of this document. An ingress RBridge selects the tree-VLAN correspondence it wishes to use from the list announced by the highest priority tree root. It SHOULD NOT transmit VLAN x frame on tree y if the highest priority tree root does not say VLAN x is allowed on tree y.

If we make sure one VLAN is allowed on one and only one tree, we can keep the number of multicast forwarding table entries on any RBridge fixed at 4K maximum (or up to 16M in case of fine grained label). Take Figure 1 as example, two trees rooted at RB1 and RB2 respectively. The highest priority tree root appoints the tree1 to carry VLAN 1-2000 and tree2 to carry VLAN 2001-4095. With such announcement by the highest priority tree root, every RBridge which understands the announcement will not send VLAN 2001-4095 traffic on tree1 and not send VLAN 1-2000 traffic on tree2. Then no RBridge would need to store the entries for tree1/VLAN2001-4095 or tree2/VLAN1-2000. Figure 2 shows the multicast forwarding table on an RBridge before and after we perform the VLAN based tree selection. The number of entries is reduced by a factor f, f being the number of trees used in the campus. In this example, it is reduced from 2*4095 to 4095. This affects both transit nodes and edge nodes. Data plane encoding does not change.

tree nickname	VLAN	port list
tree 1	1	
tree 1	2	
tree 1	...	
tree 1	...	
tree 1	...	
tree 1	4094	
tree 1	4095	
tree 2	1	
tree 2	2	
tree 2	...	
tree 2	...	
tree 2	...	
tree 2	...	
tree 2	4094	
tree 2	4095	

tree nickname	VLAN	port list
tree 1	1	
tree 1	2	
tree 1	...	
tree 1	1999	
tree 1	2000	
tree 2	2001	
tree 2	2002	
tree 2	...	
tree 2	4094	
tree 2	4095	

Figure 2. Multicast forwarding table before (left) & after (right)

3.2. Sub-TLVs for the Router Capability TLV

Four new APPsub-TLVs that can be carried in E-L1FS FS-LSPs [rfc7180bis] are defined below. They can be considered analogous to finer granularity versions of the Tree Identifiers Sub-TLV and the Trees Used Identifiers Sub-TLV in [RFC7176].

3.2.1. The Tree and VLANs APPsub-TLV

The Tree and VLANs (TREE-VLANs) APPsub-TLV is used to announce the VLANs allowed on each tree by the RBridge that has the highest priority to be a tree root. Multiple instances of this sub-TLV may be carried. The same tree nicknames may occur in the multiple Tree-VLAN RECORDs within the same or across multiple sub-TLVs. The sub-TLV format is as follows:

```
                          1 1 1 1 1 1
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |   Type = tbd1                 |         (2 bytes)
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |   Length                      |         (2 bytes)
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...-+-+
 |   Tree-VLAN RECORD (1)                        |  (6 bytes)
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...-+-+
 |   .................                           |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...-+-+
 |   Tree-VLAN RECORD (N)                        |  (6 bytes)
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...-+-+
```

where each Tree-VLAN RECORD is of the form:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Nickname                     |  (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| RESV  |        Start.VLAN               |  (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| RESV  |        End.VLAN                 |  (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

o Type: TRILL GENINFO APPsub-TLV type, set to tbd1 (TREE-VLANs).

o Length: 6*n bytes, where there are n Tree-VLAN RECORDs. Thus the value of Length can be used to determine n. If Length is not a multiple of 6, the sub-TLV is corrupt and MUST be ignored.

o Nickname: The nickname identifying the distribution tree by its root.

o RESV: 4 bits that MUST be sent as zero and ignored on receipt.

o Start.VLAN, End.VLAN: These fields are the VLAN IDs of the allowed VLAN range on the tree, inclusive. To specify a single VLAN, the VLAN’s ID appears as both the start and end VLAN. If End.VLAN is less than Start.VLAN the Tree-VLAN RECORD MUST be ignored.

3.2.2. The Tree and VLANs Used APPsub-TLV

This APPsub-TLV has the same structure as the Tree and VLANs APPsub-TLV (TREE-VLANs) specified in Section 3.2.1. The only difference is that its APPsub-TLV type is set to tbd2 (TREE-VLAN-USE), and the Tree-VLAN RECORDs listed are those the originating RBridge allows.

3.2.3. The Tree and FGLs APPsub-TLV

The Tree and FGLs (TREE-FGLs) APPsub-TLV is used to announce the FGLs allowed on each tree by the RBridge that has the highest priority to be a tree root. Multiple instances of this APPsub-TLV may be carried. The same tree nicknames may occur in the multiple Tree-FGL RECORDs within the same or across multiple APPsub-TLVs. Its format is as follows:

```
                    1 1 1 1 1 1
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Type = tbd3                 |         (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Length                      |         (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...-+-+
|   Tree-FGL RECORD (1)                     |  (8 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...-+-+
|   .................                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...-+-+
|   Tree-FGL RECORD (N)                     |  (8 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...-+-+
```

where each Tree-VLAN RECORD is of the form:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Nickname                     |         (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...-+
|            Start.FGL                          | (3 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...-+
|            End.FGL                            | (3 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...-+
```

o Type: TRILL GENINFO APPsub-TLV type, set to tbd3 (TREE-FGLs).

o Length: 8*n bytes, where there are n Tree-FGL RECORDs. Thus the value of Length can be used to determine n. If Length is not a multiple of 8, the sub-TLV is corrupt and MUST be ignored.

o Nickname: The nickname identifying the distribution tree by its root.

o RESV: 4 bits that MUST be sent as zero and ignored on receipt.

o Start.FGL, End.FGL: These fields are the FGL IDs of the allowed FGL range on the tree, inclusive. To specify a single FGL, the FGL’s

ID appears as both the start and end FGL. If End.FGL is less than Start.FGL the Tree-FGL RECORD MUST be ignored.

3.2.4. The Tree and FGLs Used APPsub-TLV

This APPsub-TLV has the same structure as the Tree and FGLs APPsub-TLV (TREE-FGLs) specified in Section 3.2.3. The only difference is that its APPsub-TLV type is set to tbd4 (TREE-FGL-USE), and the Tree-FGL RECORDs listed are those the originating RBridge allows.

3.3. Detailed Processing

The highest priority tree root RBridge MUST include all the necessary tree related APPsub-TLVs defined in [RFC7176] as usual in its E-L1FS FS-LSP and MAY include the Tree and VLANs Sub-TLV (TREE-VLANs) and or Tree and FGLs Sub-TLV (TREE-FGLs) in its E-L1FS FS-LSP [rfc7180bis]. In this way it MAY indicate that each VLAN and/or FGL is only allowed on one or some other number of trees less than the number of trees being calculated in the campus in order to save table space in the fast path forwarding hardware.

An ingress RBridge that understands the TREE-VLANs APPsub-TLV SHOULD select the tree-VLAN correspondences it wishes to use and put them in TREE-VLAN-USE APPsub-TLVs. If there were multiple tree nicknames announced in TREE-VLANs Sub-TLV for a VLAN x, ingress RBridge must choose one of them if it supports this feature. For example, the ingress RBridge may choose the closest (minimum cost) root from them. How to make such choice is out of the scope of this document. It may be desirable to have some fixed algorithm to make sure all ingress RBs choose the same tree for VLAN x in this case. Any single Data Label that the ingress RBridge is interested in should be related to one and only one tree ID in TREE-VLAN-USE to minimize the multicast forwarding table size on other RBridges but as long as the Data Label is related to less than all the trees being calculated, it will reduce the burden on the forwarding table size.

When an ingress RBridge tries to encapsulate a multi-destination frame for Data Label x, it SHOULD use the tree nickname that it selected previously in TREE-VLAN-USE or TREE-FGL-USE for Data Label x.

If RBridge RBn does not perform pruning, it builds the multicast forwarding table exactly same as that in [RFC6325].

If RBn prunes the distribution tree based on VLANs, RBn uses the information received in TREE-VLAN-USE APPsub-TLVs to mark the set of VLANs reachable downstream for each adjacency and for each related tree. If RBn prunes the distribution tree based on FGLs, RBn uses the information received in TRILL-FGL-USE APPsub-TLVs to mark the set of FLGs reachable downstream for each adjacency and for each related tree.

Logically, an ingress RBridge that does not support VLAN based tree selection is equivalent to the one that supports it and announces all the combination pair of tree-id-used and interested-vlan as TREE-VLAN-USE and correspondingly for FGL.

3.4. Failure Handling

Failure of a tree root that is not the highest priority: It is the responsibility of the highest priority tree root to inform other RBridges of any change in the allowed tree-VLAN correspondence. When the highest priority tree root learns the root of tree t fails, it should re-assign the VLANs allowed on tree t to other trees or to a tree replacing the failed one.

Failure of the highest priority tree root: It is RECOMMENDED that the second highest priority tree root be pre-configured with the proper knowledge of the tree-VLAN correspondence allowed when the highest priority tree root fails. The information announced by the second priority tree root would be stored by all RBridges but would not take effect unless the RBridge noticed the failure of the highest priority tree root. When the highest priority tree root fails, the former second priority tree root will become the highest priority tree root of the campus. When an RBridge notices the failure of the original highest priority tree root, it can immediately use the stored information announced by the original second priority tree root. It is recommended that the tree-VLAN correspondence information be pre-configured on the second highest priority tree root to be the same as that on the highest priority tree root for the trees other than the highest priority tree itself. This can minimize the change of multicast forwarding table in case of the highest priority tree root failure. For a large campus, it may make sense to pre-configure this information in a similar way on the third, fourth, or even lower priority tree root RBridges.

In some transient conditions or in case of misbehavior by the highest priority tree root, an ingress RBridge may encounter the following scenarios:

- No tree has been announced to allow VLAN x frames

- An ingress RBridge is supposed to transmit VLAN x frames on tree t, but root of tree t is no longer reachable.

For the second case, an ingress RBridge may choose another reachable tree root which allows VLAN x according to the highest priority tree root announcement. If there is no such tree available, then it is same as the first case above. Then the ingress RBridge should be 'downgraded' to a conventional BRridge with behavior as specified in [RFC6325]. A timer should be set to allow the temporary transient stage to complete before the change of responsive tree or 'downgrade' takes effect. The value of timer should at least be set to the LSP flooding time of the campus.

3.5. Multicast Extensions

Data Label based tree selection is easily extended to (Data Label + Layer 2 or 3 multicast group) based tree selection. We can appoint multicast group 1 in VLAN 10 to tree1 and appoint group 2 in VLAN 10 to tree2 for better load sharing. One additional APPsub-TLV is specified as follows:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Type = tbd5                   |  (2 byte)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Length                        |  (2 byte)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Tree Nickname             |  (2 bytes)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Group Sub-Sub-TLVs                (variable)
+-+-+-+-+-+-+-+-+-+-+....
```

o Type: TRILL GENINFO APPsub-TLV type, set to tbd5 (TREE-GROUPs).

o Length: 2 + the length of the Group Sub-Sub TLVs included

o Nickname: The nickname identifying the distribution tree by its root.

o RESV: 4 bits that MUST be sent as zero and ignored on receipt.

o Group Sub-Sub-TLVs: Zero or more of the TLV structure that are allowed as sub-TLVs of the GADDR TLV [RFC7176]. Each such TLV structure specifies a multicast group and either a VLAN or FGL. Although these TLV structure are considered sub-TLVs when they appear inside a GADDR TLV, they are technically sub-sub-TLVs when they appear inside the TREE-GROUPs APPsub-TLV.

4. Backward Compatibility

RBridges MUST include the TREE-USE-IDs and INT-VLAN sub-TLVs in their

LSPs when required by [RFC6325] whether or not they supports the new TREE-VLAN-USE or TREE-FGL-USE sub-TLVs specified by this draft.

RBridges that understand the new TREE-VLAN-USE sub-TLV sent from another RBridge RBn should use it to build the multicast forwarding table and ignore the TREE-USE-IDs and INT-VLAN sub-TLVs sent from the same RBridge. TREE-USE-IDs and INT-VLAN sub-TLVs are still useful for some purposes other than building multicast forwarding table, for example RPF table building, spanning tree root notification, etc. If the RBridge does not receive TREE-VLAN-USE sub-TLV from RBn, it uses the conventional way described in [RFC6325] to build the multicast forwarding table.

For example, there are two distribution trees, tree1 and tree2 in the campus. RB1 and RB2 are RBridges that use the new APPsub-TLVs described in this document. RB3 is an old RBridge that is compatible with [RFC6325]. Assume RB2 is interested in VLANs 10 and 11 and RB3 is interested in VLANs 100 and 101. Hence RB1 receives ((tree1, VLAN10), (tree2, VLAN11)) as TREE-VLAN-USE sub-TLV and (tree1, tree2) as TREE-USE-IDs sub-TLV from RB2 on port x. And RB1 receives (tree1) as TREE-USE-IDs sub-TLV and no TREE-VLAN-USE sub-TLV from RB3 on port y. RB2 and RB3 announce their interested VLANs in INT-VLAN sub-TLV as usual. Then RB1 will build the entry of (tree1, VLAN10, port x) and (tree2, VLAN11, port x) based on RB2's LSP and mechanism specified in this document. RB1 also builds entry of (tree1, VLAN100, port y), (tree1, VLAN101, port y), (tree2, VLAN100, port y), (tree2, VLAN101, port y) based on RB3's LSP in conventional way. The multicast forwarding table on RB1 with merged entry would be like the following.

tree nickname	VLAN	port list
tree 1	10	x
tree 1	100	y
tree 1	101	y
tree 2	11	x
tree 2	100	y
tree 2	101	y

It is expected that the table is not as small as the one where every RBridge supports the new TREE-VLAN-USE sub-TLVs. The worst case in a hybrid campus is the number of entries equal to the number in current practice which does not support VLAN based tree selection. Such an extreme case happens when the interested VLAN set from the new RBridges is a subset of the interested VLAN set from the old RBridges.

VLAN based tree selection is compatible with the current practice. Its effectiveness increases with more RBridge supporting this feature in the TRILL campus.

5. Security Considerations

This document does not change the general RBridge security considerations of the TRILL base protocol. The APPsub-TLVs specified can be secured using the IS-IS authentication feature [RFC5310]. See Section 6 of [RFC6325] for general TRILL security considerations.

6. IANA Considerations

IANA is requested to assigne five new TRILL APPsub-TLV type codes as specified in Section 3 and update the TRILL Parameters registry as shown below.

Type	Name	Reference
tbd1	TREE-VLANs	[this document]
tbd2	TREE-VLAN-USE	[this document]
tbd3	TREE-FGLs	[this document]
tbd4	TREE-FGL-USE	[this document]
tbd5	TREE-GROUPs	[this document]

7. References

7.1 Normative References

[RFC6325] Perlman, R., et.al. "RBridge: Base Protocol Specification", RFC 6325, July 2011.

[RFC6439] Eastlake, D. et.al., "RBridge: Appointed Forwarder", RFC 6439, November 2011.

[RFC7172] Eastlake 3rd, D., Zhang, M., Agarwal, P., Perlman, R., and D. Dutt, "Transparent Interconnection of Lots of Links (TRILL): Fine-Grained Labeling", RFC 7172, May 2014,

<http://www.rfc-editor.org/info/rfc7172>.

[RFC7176] Eastlake 3rd, D., Senevirathne, T., Ghanwani, A., Dutt, D., and A. Banerjee, "Transparent Interconnection of Lots of Links (TRILL) Use of IS-IS", RFC 7176, May 2014, <http://www.rfc-editor.org/info/rfc7176>.

[rfc7180bis] Eastlake 3rd, D. et. Al. draft-eastlake-trill-rfc7180bis, work in progress.

7.2 Informative References

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

8. Acknowledgments

Authors wish to thank David M. Bond, Liangliang Ma, Rakesh Kumar R for the valuable comments (names in alphabet order).

Authors' Addresses

Yizhou Li
Huawei Technologies
101 Software Avenue,
Nanjing 210012
China

Phone: +86-25-56624629
Email: liyizhou@huawei.com

Donald Eastlake
Huawei R&D USA
155 Beaver Street
Milford, MA 01757 USA

Phone: +1-508-333-2270
Email: d3e3e3@gmail.com

Weiguo Hao
Huawei Technologies
101 Software Avenue,
Nanjing 210012

China

Phone: +86-25-56623144
Email: haoweiguo@huawei.com

Hao Chen
Huawei Technologies
101 Software Avenue,
Nanjing 210012
China

Email: philips.chenhao@huawei.com

Radia Perlman
EMC
2010 256th Avenue NE, #200
Bellevue, WA 98007
USA

Email: Radia@alum.mit.edu

Naveen Nimmu
Broadcom
9th Floor, Building no 9, Raheja Mind space
Hi-Tec City, Madhapur,
Hyderabad - 500 081, INDIA

Phone: +1-408-218-8893
Email: naveen@broadcom.com

Somnath Chatterjee
Cisco Systems,
SEZ Unit, Cessna Business Park,
Outer ring road,
Bangalore - 560087
India

Email: somnath.chatterjee01@gmail.com

Sunny Rajagopalan
IBM

Email: sunny.rajagopalan@us.ibm.com