

TRILL working group
Internet Draft
Intended status: Standard Track
Expires: Sept 2013

L. Dunbar
D. Eastlake
Huawei
Radia Perlman
Intel
I. Gashinsky
Yahoo
February 22, 2013

Directory Assisted TRILL Encapsulation
draft-dunbar-trill-directory-assisted-encap-03.txt

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

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

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

This Internet-Draft will expire on August 22, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in

Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Abstract

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

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 0.

The term ''TRILL'' and ''RBridge'' are used interchangeably in this document. The term ''subnet'' and ''VLAN'' are also used interchangeably because it is very common to map one subnet to one VLAN.

Table of Contents

1. Introduction	2
2. Terminology	3
3. Directory Assistance to Non-RBridge	3
4. Source Nickname in Frames Encapsulated by Non-RBridge Nodes..	6
5. Conclusion and Recommendation.....	6
6. Manageability Considerations.....	6
7. Security Considerations.....	6
8. IANA Considerations	6
9. Acknowledgments	6
10. References	7
Authors' Addresses	7
Intellectual Property Statement.....	8
Disclaimer of Validity	9

1. Introduction

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

[RBridge-directory] describes the framework for RBridge edge to get MAC&VLAN<->RBridgeEdge mapping from a directory service in data center environment instead of flooding unknown DAs across TRILL domain. When directory is used, any node, even non-RBridge node, can

perform the TRILL encapsulation. This draft is to demonstrate the benefits of non-RBridge nodes performing TRILL encapsulation.

2. Terminology

AF Appointed Forwarder RBridge port

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

FDB: Filtering Database for Bridge or Layer 2 switch

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.

VM: Virtual Machines

3. Directory Assistance to Non-RBridge

With directory assistance [RBridge-Directory], a non-RBridge can determine if a packet needs to be forwarded across the RBridge domain. Suppose the RBridge domain boundary starts at network switches (i.e. not virtual switches embedded on servers), a directory can assist Virtual Switches embedded on servers to encapsulate proper TRILL header by providing the information of the egress RBridge edge to which the target is attached. If a target is not attached to other RBridge edge nodes based on the directory [RBridge-Directory], the non-RBridge node can forward the data frames natively, i.e. not encapsulating any TRILL header.

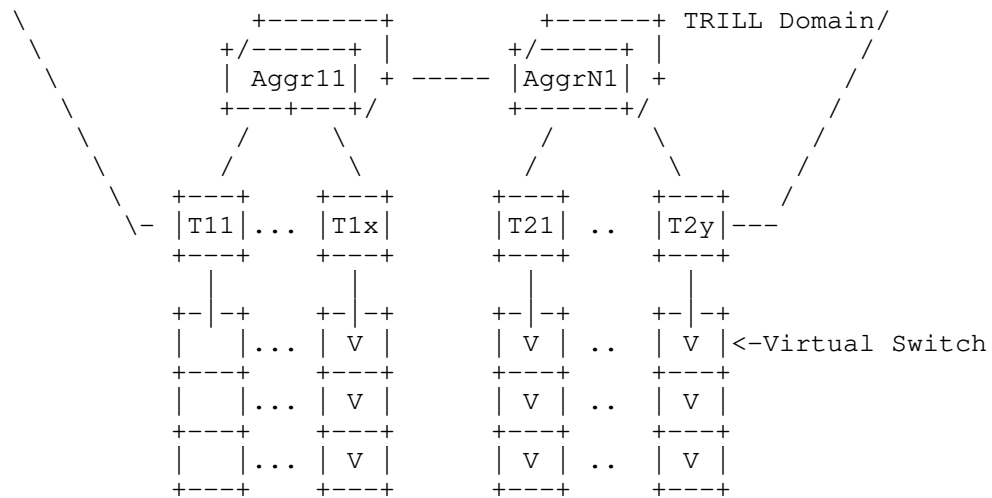


Figure 1: TRILL domain in typical Data Center Network

When a TRILL encapsulated data packet reaches the ingress RBridge, the ingress RBridge can simply forward the pre-encapsulated packet to the RBridge that is specified in the DA field of the TRILL header of the data frame. When the ingress RBridge receives a native Ethernet frame, it only forward the data frame to the directly attached bridged LAN.

Under this environment, the ingress RBridge doesn't need to flood the received Ethernet data frames to TRILL domain when the DA in the Ethernet data frames is unknown. Under this scheme, for an RBridge with multiple ports connected to a bridged LAN, data frames received from TRILL domain, decapsulated and forwarded to the bridged LAN via one port, and flooded back to the RBridge via another port, won't be encapsulated again and forwarded back TRILL domain.

That means there is no need to worry about AF ports and all RBridge edge ports connected to one bridged LAN can receive and forward pre-encapsulated traffic, which greatly improves the overall network utilization.

Note: [RBridge] 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 data frames do not need to be sent across RBridge domain, they are switched by all nodes/ports per IEEE802.1Q and RBridge edge will

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 a non-RBridge node, using the Ingress RBridge edge node's nickname in the source address field will make the ingress RBridge node receive TRILL frames with its own nickname in the frames' source address field, which can be confusing.

To avoid confusion of edge RBridges receiving TRILL encapsulated frames with their own nickname in the frames' source address field from neighboring non-RBridge nodes, a new nickname can be given to an RBridge edge node, e.g. Phantom Nickname, to represent all the TRILL Encapsulating Nodes attached to the RBridge edge node.

When the Phantom Nickname is used in the Source Address field of a TRILL frame, it is understood that the TRILL encapsulation is actually done by a non-RBridge node which is attached to an edge port of an RBridge Ingress node.

5. Conclusion and Recommendation

When directory service is available, nodes that are outside TRILL domain, i.e. don't participate in TRILL IS/IS routing protocol, become capable of encapsulating TRILL header for data frames destined for remote RBridges that is not on the same bridged LAN. The non-RBridge encapsulation approach is especially useful when there are many 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.

6. Manageability Considerations

TBD.

7. Security Considerations

TBD.

8. IANA Considerations

TBD

9. Acknowledgments

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

10. References

[RBridge-Directory] Dunbar, et, al ''TRILL (Transparent Interconnection of Lots of Links) Edge Directory Assistance Framework'', <draft-ietf-trill-directory-framework-03>, March, 2013

[RBridges] Perlman, et, al ''RBridge: Base Protocol Specification'', <draft-ietf-trill-rbridge-protocol-16.txt>, March, 2010

[RBridges-AF] Perlman, et, al ''RBridges: Appointed Forwarders'', <draft-ietf-trill-rbridge-af-02.txt>, April 2011

[ARMD-Problem] Dunbar, et, al, ''Address Resolution for Large Data Center Problem Statement'', Oct 2010.

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

Authors' Addresses

Linda Dunbar
Huawei Technologies
1700 Alma Drive, Suite 500
Plano, TX 75075, USA
Phone: (972) 543 5849
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
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

Intellectual Property Statement

The IETF Trust takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in any IETF Document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights.

Copies of Intellectual Property disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement any standard or specification contained in an IETF Document. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

All IETF Documents and the information contained therein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION THEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.

INTERNET-DRAFT
Intended status: Proposed Standard

Linda Dunbar
Donald Eastlake
Huawei
Radia Perlman
Intel
Igor Gashinsky
Yahoo
Yizhou Li
Huawei
February 25, 2013

Expires: August 24, 2012

TRILL: Directory Assistance Mechanisms
<draft-dunbar-trill-scheme-for-directory-assist-04.txt>

Abstract

This document describes optional mechanisms for using directory server(s) to assist TRILL (Transparent Interconnection of Lots of Links) edge switches in reducing multi-destination traffic, particularly ARP/ND and unknown unicast flooding.

Status of This Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

Distribution of this document is unlimited. Comments should be sent to the TRILL working group mailing list.

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/lid-abstracts.html>. The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

Table of Contents

1. Introduction.....	3
1.1 Terminology.....	3
1.2 Circumstances Causing Directory Use.....	4
2. Push Model Directory Assistance Mechanisms.....	5
2.1 Requesting Push Service.....	5
2.2 Actions by Push Directory Servers.....	5
2.3 Additional Push Details.....	6
3. Pull Model Directory Assistance Mechanisms.....	8
3.1 Pull Directory Request Format.....	8
3.2 Pull Directory Response Format.....	10
3.3 Pull Directory Hosted on an End Station.....	12
3.4 Pull Directory Request Errors.....	14
3.5 Cache Consistency.....	15
3.6 Additional Pull Details.....	17
4. Directory Use Strategies and Push-Pull Hybrids.....	18
4.1 Strategy Configuration.....	18
5. The Interface Addresses APPsub-TLV.....	21
5.1 Format of the Interface Addresses APPsub-TLV.....	21
5.2 IA-APPsub-TLV sub-sub-TLVs.....	24
5.2.1 AFN Size sub-sub-TLV.....	25
5.2.2 Fixed Address sub-sub-TLV.....	26
5.2.3 Data Label sub-sub-TLV.....	26
5.2.4 Topology sub-sub-TLV.....	27
6. Security Considerations.....	28
7. IANA Considerations.....	29
7.1 ESADI-Parameter Bits.....	29
7.2 RBridge Channel Protocol Number.....	29
7.3 Pull Directory and No Data Bits.....	29
7.4 Additional AFN Number Allocation.....	30
7.5 IA APPsub-TLV Sub-Sub-TLVs SubRegistry.....	30
8. Acknowledgments.....	32
9. References.....	33
9.1 Normative References.....	33
9.2 Informational References.....	34

1. Introduction

[DirectoryFramework] describes a high level framework for using directory servers to assist TRILL [RFC6325] edge nodes to reduce multi-destination ARP/ND and unknown unicast flooding traffic. Because multi-destination traffic becomes an increasing burden as a network scales, reducing ARP/ND and unknown unicast flooding improves TRILL network scalability. This document describes optional specific mechanisms for directory servers to assist TRILL edge nodes.

The information held by the directories is address mapping information. Most commonly, what MAC address corresponds to an IP address within a Data Label (VLAN or FGL (Fine Grained Label [RFCfgl])) and what egress TRILL switch (RBridge) that MAC address is attached to. But it could be what IP address corresponds to a MAC address or possibly other mappings. In the data center environment, it is common for orchestration software to know and control where all the IP addresses, MAC address, and VLANs/tenants are. Thus such orchestration software is appropriate for providing the directory function or for supplying the Directory(s) with information they need.

Directory services can be offered in a Push or Pull mode. Push mode, in which a directory server pushes information to RBridges indicating interest, is specified in Section 2. Pull mode, in which an RBridge queries a server for the information it wants, is specified in Section 3. Hybrid Push/Pull modes of operation are discussed in Section 4.

The mechanisms used to keep the mappings held by different Directories synchronized is beyond the scope of this document.

1.1 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 additional acronyms and terms:

Data Label: VLAN or FGL.

FGL: Fine Grained Label [RFCfgl].

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.

RBridge: An alternative name for a TRILL switch.

TRILL switch: An alternative name for an RBridge.

1.2 Circumstances Causing Directory Use

While an RBridge can consult Directory information whenever it wants, by searching through information that has been pushed to it or requesting information from a pull directory, the following are expected to be the most common circumstances leading to directory use. All of these involve cases of ingressing a native frame.

- o Ingressing an frame with an unknown unicast destination MAC. The mapping from the destination MAC and Data Label to its egress RBridge of attachment is needed to ingress the frame as unicast. If the egress RBridge is unknown, the frame must be dropped or ingressed as a multi-destination frame and flooded to all edge RBridges for its Data Label.
- o Ingressing an ARP [RFC826]. ...TBD
- o Ingressing a ND [RFC903]. ...TBD... Secure Neighbor Discovery messages [] will, in general, have to be sent to the neighbor intended so that neighbor can sign the answer; however, directory information can be used to unicast the ND packet rather than multicasting it.
- o Ingressing a RARP [RFC4861]. ...TBD

2. Push Model Directory Assistance Mechanisms

In the Push Model, Push Directory servers push down the mapping information for the various addresses of end stations in some Data Label. A Push Directory advertises whether or not it believes it is pushing complete mapping information for a Data Label. The Push Model uses the [ESADI] protocol.

With this model, it is RECOMMENDED that complete address mapping information for a Data Label be pushed and that a participating RBridge simply drop a data packet, instead of flooding the packet, if the destination unicast MAC address is in a category being pushed and can't be found in the mapping information available. This will minimize flooding of packets due to errors or inconsistencies but is not practical if directories have incomplete information.

2.1 Requesting Push Service

In the Push Model, it is necessary to have a way for an RBridge to request information from the directory server(s). RBridges simply use the ESADI protocol mechanism to announce, in the IS-IS link state database, all the Data Labels for which they are participating in [ESADI]. They are then pushed the mapping information for all such Data Labels being served by a Push Directory server.

2.2 Actions by Push Directory Servers

Push Directory servers advertise their availability to push the mapping information for a particular Data Label to ESADI participants for that Data Label by turning on a flag bit in their ESADI Parameter APPsub-TLV [ESADI] (see Section 7.1).

Each Push Directory server MUST participate in ESADI for the Data Labels for which it can push mappings and set the PD bit in their 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 RBridge that is a Push Directory server is configured with a number in the range 1 to 8, which defaults to 2, as to the number of copies it believes should be pushed. Each Push Directory server also has a priority that is its 6-byte IS-IS System ID treated as an unsigned integer where larger magnitude means higher priority.

For each Data Label it can serve, each Push Directory RBridge server orders the Push Directory servers that it can see as data reachable

[RFCclear] in the ESADI link state database for that Data Label and determines its position in that order. If a Push Directory server believes that N copies of the mappings for a Data Label should be pushed and finds that it is first in priority or, more generally, not lower than Nth in priority, it is Active. If it finds that it is N+1st or lower in priority, it is Passive.

For example, assume four Push Directory servers for Data Label X: server A with priority 123 configured to believe there should be 2 copies pushed; server B, priority 88, 1 copy; server C, priority 40, 3 copies; and server D, priority 7, 2 copies. Server A, seeing that is highest priority, is Active. Server B, seeing that it is 2nd highest priority and believing that only 1 copy should be pushed, is Passive. Server C sees that it is 3rd highest priority and believes 3 copies should be pushed, so it is Active. And server D sees it is 4th highest priority and, believing that only 2 copies should be pushed, is Passive.

If a Push Directory server is Active for Data Label X, it includes the Data Label X directory mappings it has in its ESADI-LSP for Data Label X and updates that information as the mappings it knows change. If the Push Directory server is configured to believe it has complete mapping information for Data Label X then, after it has actually transmitted all of its ESADI-LSPs for X it waits its CSNP time (see Section 6.1 of [ESADI]), and then updates its ESADI-Parameters APPsub-TLV to set the Complete Push (CP) bit to one. It then maintains the CP bit as one as long as it is Active.

If a Push Directory server is Passive for Data Label X, it removes or continues to leave out all Data Label X directory mappings it holds from its ESADI-LSP for Data Label X. However, if it was Active and was advertising the CP bit as one in its ESADI-Parameters APPsub-TLV, it first updates the CP bit to zero and sends its updated ESADI-LSP fragment zero and then waits its CSNP time before withdrawing all its directory mapping information.

2.3 Additional Push Details

Push Directory mappings can be distinguished for any other data distributed through ESADI because mappings are distributed only with the Interface Addresses APPsub-TLV specified in Section 5 and are flagged as being Push Directory data.

RBridges, whether or not they are a Push Directory server, MAY advertise any locally learned MAC attachment information in ESADI 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 would generally not be done as it

should not add anything and would just waste bandwidth and ESADI link state space. An exception would be when an RBridge learns local MAC connectivity and that information appears to be missing from the directory mapping. In that case, it SHOULD advertise the missing information unless configured not to.

Because a Push Directory server may need to advertise interest in Data Labels even though it does not want to receive user data in those Data Labels, the No Data flag bit is provided as discussed in Section 7.3.

If an RBridge notices that a Push Directory server is no longer data reachable [RFCclear], it MUST ignore any Push Directory data from that server because it is no longer being updated and may be stale.

There may be transient 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 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.

3. Pull Model Directory Assistance Mechanisms

In the Pull Model, an RBridge pulls mapping information from an appropriate Directory Server when needed.

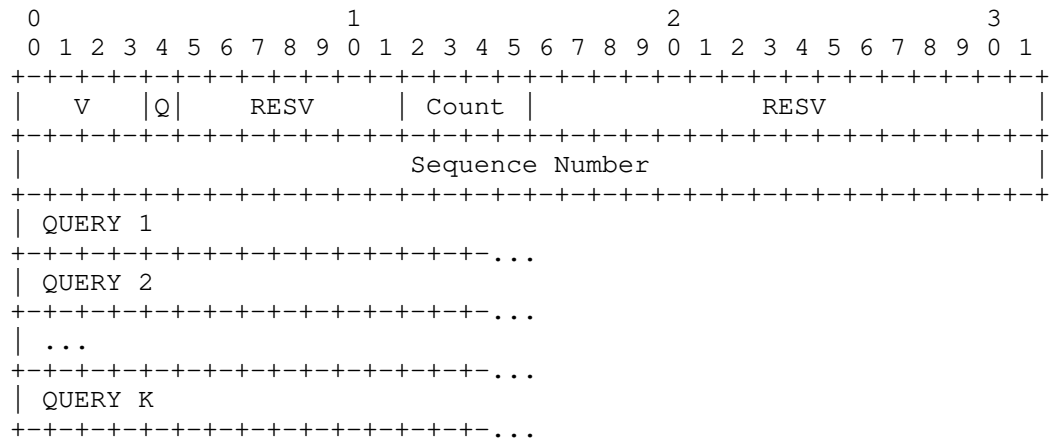
Pull Directory servers for a particular Data Label X are located by looking in the main TRILL IS-IS link state database for RBridges that advertise themselves by having the Pull Directory flag on in their Interested VLANs or Interested Labels sub-TLV [rfc6326bis] for X. If multiple RBridges indicate that they are Pull Directory Servers for a particular Data Label a pull request can be sent to any of them that is data reachable but it is RECOMMENDED that pull requests be sent to server that is least cost from the requesting RBridge.

Pull Directory requests are sent by enclosing them in an RBridge Channel [Channel] message using the Pull Directory channel protocol number (see Section 7.2). Responses are returned in an RBridge Channel message using the same channel protocol number.

The requests to Pull Directory Servers are derived from normal ARP [RFC826], ND [RFC4861], RARP [RFC903] messages or data frames with unknown unicast destination MAC addresses intercepted by the RBridge when they would otherwise be ingressed. 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 or the requester is administratively prohibited from receiving the data or the like. Thus both positive responses with data and negative responses can be cached and used for immediate response to ARP, ND, RARP, or unknown destination MAC frames, until they expire. If information previously pulled is about to expire, an RBridge MAY try to refresh it by issued a new pull request but, to avoid unnecessary requests, SHOULD NOT do so if it has not been recently used.

3.1 Pull Directory Request Format

A Pull Directory request is sent as the Channel Protocol specific content of an inter-RBridge Channel message TRILL Data packet. The Data Label in the packet is the Data Label in which the address is being looked up. The priority of the channel message is a mapping of the priority frame being ingressed that caused the request with the default mapping depending, per Data Label, on the strategy (see Section 4). The Channel Protocol specific data is formatted as follows:



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

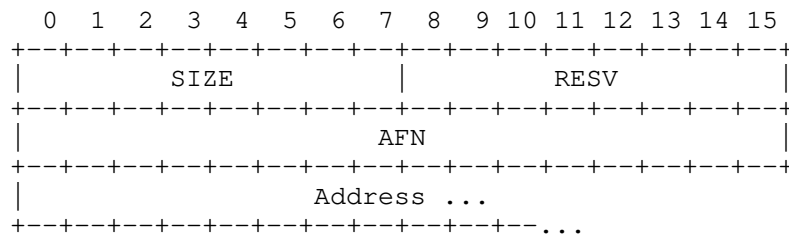
Q: Query/Response Bit. MUST be one for a query.

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

Count: Number of queries present.

Sequence Number: An opaque 32-bit quantity set by the sending RBridge, returned in any responses, and used to match up responses with queries.

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



SIZE: Size of the query data in bytes. This is the length of the Address plus 4.

RESV: A reserved byte. MUST be sent as zero and ignored on receipt.

AFN: Address Family Number of the Address.

Address: This is the address for which the query is asking for

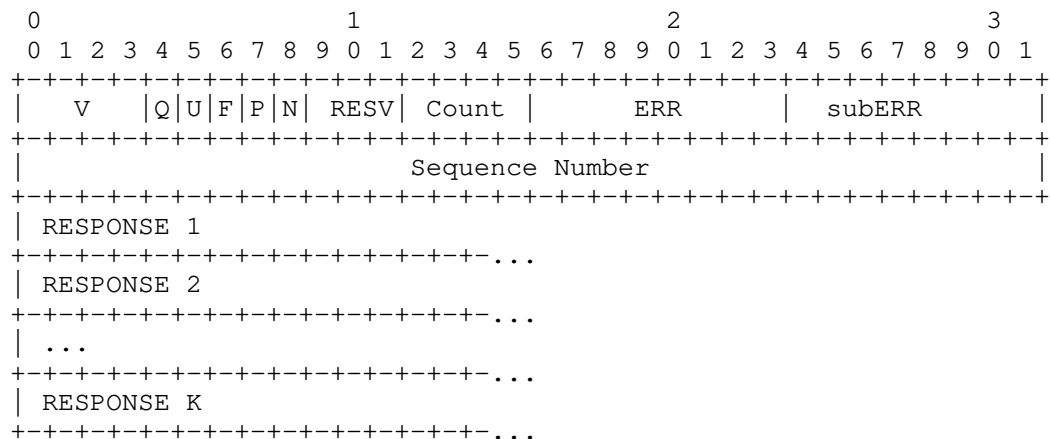
mappings. Typically it would be either (1) a MAC address, in which case the querying RBridge is interested in the RBridge by which that MAC address is reachable, or (2) an IP address, in which case the querying RBridge is interested in the corresponding MAC address and the RBridge by which that MAC address is reachable.

A query count of zero is explicitly allowed, for the purpose of pinging a Pull Directory server to see if it is responding to requests. It results in a response message that also has a count of zero.

If no response is received to a Pull Directory request within a configurable timeout, the request should be re-transmitted with the same Sequence Number up to a configurable number of times that defaults to three. If there are multiple queries in a request, responses can be received to various subsets of these queries by the timeout. In that case, the remaining unanswered queries should be re-sent in a new query with a new sequence number. If an RBridge is not capable of handling partial responses to requests with multiple queries, it MUST NOT send a request with more than one query in it.

3.2 Pull Directory Response Format

Pull Directory responses are sent as the Channel Protocol specific content of inter-RBridge Channel message TRILL Data packets. Responses are sent with the same Data Label and priority as the request to which they correspond except that the response priority is limited. This priority limit is configurable at a per RBridge level and defaults to priority 6. The Channel protocol specific data format is as follows:



- V: Version of the Pull Directory protocol. Version zero is specified in this document.
- Q: Query/Response Bit. MUST be zero for a response.
- U: Unsolicited Bit. MUST be zero for a response to a query and one for an unsolicited "response" sent to maintain cache consistency (see Section 3.5).
- F: The Flood bit. If zero, the reply is to be unicast to the provided Nickname. If U=1, F=1 is used to flood messages for certain unsolicited cache consistency maintenance messages from an end station Pull Directory server as discussed in Section 3.5. If U=0, F is ignored.

P, N: Flags used in connection with certain flooded unsolicited cache consistency maintenance messages. Ignored if U is zero. If the P bit is a one, the solicited response message relates to cached positive response information. If the N bit is a one, the unsolicited messages related to cached negative information. See Section 3.5.

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

Count: Count is the number of responses present in the particular response message.

ERR, subERR: A two part error code. See Section 3.4.

Sequence Number: An opaque 32-bit quantity set by the requesting RBridge and copied by the Pull Directory into all responses to the query. For an unsolicited "response", the contents are unspecified.

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           |   RESV   |   Index   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|           Lifetime           |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|           Response Data ...
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

SIZE: Size of the response data in bytes plus 4.

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

Index: The relative index of the query in the request message to which this response corresponds. The index will always be one for request messages containing a single query. The index will always be zero for unsolicited "response" messages.

Lifetime: The length of time for which the response should be considered valid in seconds.

Response Data: There are two types of response data. If the ERR field is non-zero, the response data is a copy of the query data, that is, an AFN followed by an address. If the ERR field is zero, the response data is the contents of an Interface Addresses APPsub-TLV (see Section 5) without the usual TRILL GENINFO TLV type and length and without the usual IA APPsub-TLV type and length before it.

Multiple response records can appear in a response message with the same index if the answer to a query consists of multiple Interface Address APPsub-TLV contents. This would be necessary if, for example, a MAC address within a Data Label appears to be reachable by multiple R Bridges.

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.

See Section 3.4 for a discussion of how errors are handled.

3.3 Pull Directory Hosted on an End Station

Optionally, a Pull Directory actually hosted on an end station MAY be supported. In that case, when the R Bridge advertising itself as a Pull Directory server receives a query, it modifies the inter-R Bridge Channel message received into a native R Bridge Channel message and forwards it to that end station. Later, when it receives one or more responses from that end station by native R Bridge Channel messages, it modifies them into inter-R Bridge Channel messages and forwards them to the source R Bridge of the query.

The native R Bridge Channel Pull Directory messages use the same Channel protocol number as do the inter-R Bridge Pull Directory Channel messages. The native messages MUST be sent with an Outer.VLAN tag which give the priority of each message which is the priority of the original inter-R Bridge request packet. The Outer.VLAN ID used is the Designated VLAN on the link.

The native RBridge Channel message protocol dependent data for a Pull Directory query is formatted 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
+-----+-----+-----+-----+-----+-----+-----+-----+
|  V   | Q |   RESV   | Count |           Nickname           |
+-----+-----+-----+-----+-----+-----+-----+-----+
| Data Label ... (4 or 8 bytes) |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Sequence Number                |
+-----+-----+-----+-----+-----+-----+-----+-----+
| QUERY 1 |
+-----+-----+-----+-----+-----+-----+-----+-----+
| QUERY 2 |
+-----+-----+-----+-----+-----+-----+-----+-----+
| ... |
+-----+-----+-----+-----+-----+-----+-----+-----+
| QUERY K |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Data Label: The Data Label of the original inter-RBridge Pull Directory Channel protocol messages that was mapped to this native channel message. The format is the same as it appears right after the Inner.MacSA of the original Channel message.

Nickname: The nickname of the requesting RBridge.

All other fields are as specified in Section 3.1.

The native RBridge Channel message protocol specific content for a Pull Directory response is formatted 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
+-----+-----+-----+-----+-----+-----+-----+-----+
|  V   | Q | U | F | P | N | RESV | Count |       ERR       | subERR |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Nickname                        |
+-----+-----+-----+-----+-----+-----+-----+-----+
| Data Label ... (4 or 8 bytes) |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Sequence Number                |
+-----+-----+-----+-----+-----+-----+-----+-----+
| RESPONSE 1 |
+-----+-----+-----+-----+-----+-----+-----+-----+
| RESPONSE 2 |
+-----+-----+-----+-----+-----+-----+-----+-----+
| ... |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

```
| RESPONSE K
+--+--+--+--+--+--+--+--+--+--+--+--+...
```

Data Label: The Data Label to which the response applies. The format is the same as it appears right after the Inner.MacSA in TRILL Data messages.

Nickname: The nickname of the destination RBridge or, if F=1, ignored.

All other fields are as specified in Section 3.2.

3.4 Pull Directory Request Errors

An error response message is indicated by a non-zero ERR field.

If there is an error that applies to the entire request 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 expanded with a zero Lifetime and the insertion of the Index field echoed back in the response records.

If errors occur at the query level, they MUST be reported in a response message separate from the results of any successful queries. If multiple queries in a request have different errors, they MUST be reported in separate response messages. If multiple queries in a request have the same error, this error response MAY be reported in one response message.

In an error response message, the query or queries 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.

ERR values 1 through 63 are available for encoding request message level errors. ERR values 64 through 255 are available for encoding query 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 V field value
2	Request data too short
3	Administratively prohibited
4-31	(Available for allocation by Standards Action)
32	Unknown AFN
33	No mapping found
34	Administratively prohibited
35-255	(Available for allocation by Standards Action)

More TBD...?

3.5 Cache Consistency

Pull Directories **MUST** take action to minimize the amount of time that an RBridge will continue to use stale information from the Pull Directory.

A Pull Directory server **MUST** maintain one of the following, in order of increasing specificity.

1. An overall record per Data Label of when the last returned query data will expire at a requestor and when the last query record specific negative response will expire.
2. For each unit of data (IA APPsub-TLV Address Set) held by the server and each address about which a negative response was sent, when the last expected response with that unit or negative response will expire at a requester.
3. For each unit of data held by the server and each address about which a negative response was sent, a list of RBridges that were sent that unit as the response or sent a negative response to the address, with the expected time to expiration at each of them.

A Pull Directory server may have a limit as to how many RBridges it can maintain expiry information for 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 querying RBridge, the Pull Directory MUST send an unsolicited message as discussed below.

If method 1, the most crude method, is being followed, then when any information in a Data Label is changed or deleted or an additional administrative Pull Directory access restriction imposed, and there are outstanding cached positive query data response(s), an all-addresses flush positive message is flooded (multicast) within that Data Label. And if data is added or an administrative restriction is removed and there are outstanding cached negative responses, an all-addresses flush negative message is flooded. "All-addresses" is indicated by the Count in an unsolicited response being zero. On receiving an all-addresses flooded flush positive message from a Pull Directory server it has used, indicated by the U, F, and P bits being one, an RBridge discards all cached data responses it has for that Data Label. Similarly, on receiving an all addresses flush negative message, indicated by the U, F, and N bits being one, it discards all cached negative responses for that Data Label. A combined flush positive and negative can be flooded by having all of the U, 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 an RBridge floods address specific update positive unsolicited responses when data which is cached by a querying RBridge is changed or deleted or an administrative restriction is added to such data and floods an address specific update negative unsolicited responses when such information is deleted or an administrative restriction is removed from such data. Such messages are similar to the method 1 flooded unsolicited flush messages. The U and F bits will be one and the message will be multicast. However that 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 message, if it is positive the addresses in the response records in the unsolicited response are compared to the addresses about which the recipient RBridge is holding cached positive information and, if they match, the cached information is updated and its remaining cache life set to the minimum of its previous value in the cache and the Lifetime value in the unsolicited response. In the case of a newly imposed administrative restriction, the Lifetime in the unsolicited response is set to zero so the cached information immediately expired. On receiving an address specific unsolicited negative response, the addresses in the response records in the unsolicited response are compared to the addresses about which the recipient RBridge is holding cached negative information and, if they match, the cached negative information is discarded.

If method 3 is being followed, the same sort of messages are sent as with method 2 except they are not flooded but unicast only to the specific RBridges the server believes may be holding the cached

positive or negative information that may need updating.

3.6 Additional Pull Details

If an RBridge notices that a Pull Directory server is no longer data reachable [RFCclear], it MUST discard all responses it is retaining from that server within one second as the RBridge can no longer receive cache consistency 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 user data in those Data Labels, the No Data flag bit is provided as discussed in Section 7.3.

4. Directory Use Strategies and Push-Pull Hybrids

For some edge nodes which have great number of Data Labels enabled, managing the MAC&Label <-> RBridgeEdge 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 RBridge Edge 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 when hosts in specific Data Labels, say VLANs 1 through 100, communicate regularly with external peers, 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 once a day (or once a few days) 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 RBridges 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.

4.1 Strategy Configuration

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

0. No directory use. The RBridge 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, including ARP, ND, RARP, and unknown MAC destination addresses, are flooded.
1. Use Push only. The RBridge subscribes to Push Directory data for Data Label X.
2. Use Pull only. When the RBridge ingresses a frame in Data Label X that can use Directory information, if it has cached positive

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 RBridge subscribes to Push Directory data for Data Label X. When it ingresses a frame in Data Label X that can use Directory information,

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 positive Pull Directory information to handle the frame given the major mode, the directory information is simply used and, in that instance, the minor modes 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. Otherwise, the same as A.
- D. Discard if complete or Flood immediate. If you have complete Push Directory information and the address is not in that information, discard the frame. Otherwise, the same as B.

In addition, the Pull Directory priority for an Pull Directory requests sent can be configured on a per Data Label, per ingressed frame priority basis. The default mappings are as follows:

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 network connectivity and so is avoided by default for directory

traffic.

5. The Interface Addresses APPsub-TLV

[[[This Section 5 is fairly long and complex. Should it be a separate document?]]]

This section specifies a TRILL APPsub-TLV that enables the convenient representation of sets of addresses of different types such that all of the addresses in each set designate the same end station interface (port). For example, an EUI-48 MAC (Extended Unique Identifier 48-bit, Media Access Control [RFC5342]) address, IPv4 address, and IPv6 address can be reported as all three corresponding to the same interface. This APPsub-TLV is used inside the TRILL GENINFO TLV as specified in [ESADI] and the value portion is used inside Pull Directory responses as specifies in Section 3.

Although, in some IETF protocols, address field types are represented by EtherType [RFC5342] or Hardware Type [RFC5494] only Address Family Number is used in this APPsub-TLV.

5.1 Format of the Interface Addresses APPsub-TLV

The Interface Addresses APPsub-TLV is used to indicate that a set of addresses indicate the same end-station interface and to associate that interface with the TRILL switch by which the interface is reachable. These addresses can be in different address families. For example, it can be used to declare that an end-station interface with a particular IPv4 address, IPv6 address, and EUI-48 MAC address is reachable from a particular TRILL switch.

The Template field value indicates certain well known sets of addresses or gives the number of AFNs following. When AFNs are listed, the set of AFNs provides a template for the type and order of addresses in each Address Set.

```

+-----+
| Type = TBD | (1 byte)
+-----+
| Length | (1 byte)
+-----+
| Nickname | (2 bytes)
+-----+
| Flags | (1 byte)
+-----+
| Confidence | (1 byte)
+-----+
| Addr Set End | (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 TBD[#2 suggested] (IA-SUBTLV).
- o Length: Variable, minimum 5. If length is 4 or less, the APPsub-TLV MUST be ignored.
- o Nickname: The nickname of the RBridge by which the address sets are reachable.
- o Flags: A byte of flags as follows:

```

0 1 2 3 4 5 6 7
+-----+
| D | L |   Resv   |
+-----+

```

D: If D is one, the APPsub-TLV contains Push Directory information.

L: If L is one, the APPsub-TLV contains information learned locally by observing ingress frames. (Both D and L can be one in the same APPsub-TLV.)

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

- o Confidence: This 8-bit quantity indicates the confidence level in the addresses being transported [RFC6325].
- o Addr Set End: The unsigned offset of the byte, within the TLV value part, of the last byte of the last Address Set. This will be the byte just before the first sub-TLV if any sub-TLVs are present. [RFC5305]
- o Template: The initial byte of this field is the unsigned integer K. If K has a value from 1 to 63, it indicates that this initial byte is followed by a list of K AFNs (Address Family Numbers) in the template specifying the structure and order of each Address Set occurring later in the TLV. The minimum valid value is 1. If K is 64 to 255, it indicates that the Template for each Address Set is a specific well known Template. If the Template includes explicit AFNs, they look like the following.

```

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

```

- o AFN: A two-byte Address Family Number. The number of AFNs present is given in first byte of the Template field if that value is less than 64. This 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 IPv4 and EUI-48, in that order, then each Address set present will consist of a 4-byte IPv4 address followed by a 6-byte MAC 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-TLV as specified below, the receiving IS will be unable to parse the Address Sets and MUST ignore the enclosing TLV.
- o Address Set: Each address set consists of a sequence of addresses of the types given by the Template earlier in the 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 Address Sets. The Address Sets must fit within the TLV. If the product of the size of an Address Set and the number of Address Sets is so large that this is not true, the APPsub-TLV

is ignored.

- o sub-sub-TLVs: If the Address Sets indicated by Addr Sets End do not completely fill the Length of the TLV, the remaining bytes are parsed as sub-sub-TLVs [RFC5305]. Any such sub-sub-TLVs that are not known to the receiving RBridge are ignored. Should this 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 5.2.

Different IA-APPsub-TLVs within the same or different EADI-LSPs or Pull Directory response from the same RBridge may have different Templates. The same AFN may occur more than once in a Template and the same address may occur in more than one address set. For example, an EUI-48 MAC address interface might have three 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 EUI-48 and one IPv6 address could be used in an IA-APPsub-TLV with three address sets each having the same EUI-48 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 RBridge must know at least the size of the address each AFN in the Template specifies; however, the presence of the Addr Set End field means that the sub-TLVs, if any, can always be located by a receiving IS. An RBridge can be assumed to know the size of IPv4 and IPv6 addresses (AFNs 1 and 2) and the size of the additional AFNs allocated by the IANA Considerations below. Should an RBridge wish to include an AFN that some receiving RBridge in the campus may not know, it SHOULD include an AFN-Size sub-sub-TLV as described below. If an IA-APPsub-TLV is received with one or more AFNs in its template for which the receiving RBridge does not know the length and for which an AFN-Size sub-sub-TLV is not present, that IA-APPsub-TLV will be ignored.

5.2 IA-APPsub-TLV sub-sub-TLVs

IA-APPsub-TLVs may 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.

5.2.1 AFN Size sub-sub-TLV

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

1. One or more AFNs that are unknown to the receiving RBridge appears in the template. If an AFN Size sub-sub-TLV is present for each such AFN, the at least the IA-APPsub-TLV can be parse the Address Sets and make use of any address types present that it does understand.
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-APPsubTLV.

```

+-----+-----+
| Type = AFNsz | (1 byte)
+-----+-----+
| Length       | (1 byte)
+-----+-----+
| AFN Size Record(s) | (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 n 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 of the address specified by the AFN field.

This sub-sub-TLV may occur multiple times in an enclosing IA-APPsub-TLV.

An AFN Size sub-sub-TLV for any AFN known to the receiving RBridge (which always includes AFN 1 and 2 and the AFNs specified in xxx) is compared with the size known to the RBridge and if they differ, the IA-APPsub-TLV is ignored.

5.2.2 Fixed Address sub-sub-TLV

There may be cases where, in an Interface Addresses TLV, the same address would appear across every address set in the TLV. To avoid having a larger template and wasted space in all Address Sets, this sub-sub-TLV can be used to indicate such a fixed address

```

+-----+
|Type=FIXEDADR| (1 byte)
+-----+
| Length      | (1 byte)
+-----+
| AFN         | (2 bytes)
+-----+
| Fixed Address (variable)
+-----+

```

- o Type: Data Label sub-sub-TLV type, set to 2 (FIXEDADR).
- o Length: variable, minimum 3. If Length is 2 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.

5.2.3 Data Label sub-sub-TLV

When used with Push or Pull Directories, the Data Label is indicated by the Data Label of the ESADI instance (Push) or RBridge Channel message (Pull) in which the IA APPsub-TLV appears and any occurrence of this sub-sub-TLV is ignored. However, the IA APPsub-TLV might be used in other contexts where this sub-sub-TLV indicates the Data Label of the Address Sets and multiple occurrences of this sub-sub-TLV indicate that the Address Sets exist in all of the Data Labels.

```

+-----+
|Type=DATALEN| (1 byte)
+-----+
| Length| (1 byte)
+-----+
| Data Label| (variable)
+-----+

```

- o Type: Data Label sub-TLV type, set to 3 (DATALEN).
- o Length: 2 or 3
- 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 [RFCfgl].

5.2.4 Topology sub-sub-TLV

The presence of this sub-sub-TLV indicates that the Address Sets are in the topology give. If it occurs multiple times, then the Address Sets are in all of the topologies listed.

```

+-----+
|Type=DATALEN| (1 byte)
+-----+
| Length| (1 byte)
+-----+
| RESV | Topology| (2 bytes)
+-----+

```

- o Type: Data Label sub-TLV type, set to 3 (DATALEN).
- o Length: 2.

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

- o Topology: The 12-bit topology number.

6. Security Considerations

Push Directory data is distributed through ESADI-LSPs [ESADI] which can be authenticated with the same mechanisms as IS-IS LSPs. See [RFC5304] and [RFC5310].

Pull Directory queries and responses are transmitted as RBridge-to-RBridge or native RBridge Channel messages. Such messages can be secured by TBD

For general TRILL security considerations, see [RFC6325].

7. IANA Considerations

This section give IANA allocation and registry considerations.

7.1 ESADI-Parameter Bits

IANA is request to allocate two ESADI-Parameter TRILL APPsub-TLV flag bits for "Push Directory" and "Complete Push" and to create a sub-registry in the TRILL Parameters Registry as follows:

Sub-Registry: ESADI-Parameter APPsub-TLV Bits

Registration Procedures: IETF Review

References: [ESADI], This document

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

7.2 RBridge Channel Protocol Number

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

7.3 Pull Directory and No Data Bits

IANA is requested to allocate two currently reserved bits in the Interested VLANs field of the Interested VLANs sub-TLV (suggested bits 3 and 4) and the Interested Labels field of the Interested Labels sub-TLV (suggested bits 5 and 6) [rfc6326bis] to indicate Pull Directory server (PD) and No Data (ND) respectively. These bits are to be added to the subregistry set up in [ESADI].

In the TRILL base protocol [RFC6325] as extended for FGL [rfcFGL], the mere presence of an Interested VLANs or Interested Labels sub-TLVs in the LSP of an RBridge indicates connection to end stations in the VLANs or FGLs listed and thus a desire to receive multi-destination traffic in those Data Labels although multicast traffic

might be pruned. But, with Push and Pull Directories, advertising that you are a directory server requires using these sub-TLVs as part of advertising that you are a directory server. If such a directory server does not wish to receive multi-destination user data for the Data Labels it lists in one of these sub-TLVs, it sets the "No Data" (ND) bit to one. This means that data on a distribution tree may be pruned so as not to reach the "No Data" RBridge as long as there are no RBridges interested in the Data who are beyond the "No Data" RBridge. This bit is backwards compatible as RBridges ignorant of it will simply not prune when it could, which is safe but may cause increased link utilization.

7.4 Additional AFN Number Allocation

IANA is requested to allocate four new AFN numbers as follows:

Number	Description	References	-----	-----
TBD(26)	EUI-48	RFC 5342, this document		
TBD(27)	OUI	RFC 5342, this document		
TBD(28)	MAC/24	This document.		
TBD(29)	IPv6/64	This document.		

The OUI AFN is provided so that MAC addresses can be abbreviated if they have the same upper 24 bits. In particular, if there is an OUI provided as a Fixed Address sub-sub-TLV (see Section 5.2.2) then, whenever a MAC/24 address appears within an Address Set (as indicated by the Template), the OUI is used as the first 24 bits of the actual MAC address for the Address Set.

MAC/24 is a 24-bit suffixes intended to be pre-fixed by an OUI as in the previous paragraph. In absence of an OUI specified as a Fixed Address in the same APPsub-TLV, the Address Set cannot be used.

IPv6/64 is an 8-byte quantity that is the first 64 bits of an IPv6 address. If present, there will normally be an EUI-64 address in the address set to provide the lower 64 bits of the IPv6 address. For this purpose, an EUI-48 is expanded to 64 bits as described in [RFC5342].

7.5 IA APPsub-TLV Sub-Sub-TLVs SubRegistry

IANA is requested to establish a new subregistry 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: IETF Review

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	This document
255	Reserved	

8. Acknowledgments

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

9. References

Normative and Informational References are given below.

9.1 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
- [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.
- [RFC5305] - Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, October 2008.
- [RFC5342] - Eastlake 3rd, D., "IANA Considerations and IETF Protocol Usage for IEEE 802 Parameters", BCP 141, RFC 5342, September 2008.
- [RFC5494] - Arkko, J. and C. Pignataro, "IANA Allocation Guidelines for the Address Resolution Protocol (ARP)", RFC 5494, April 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.
- [rfc6326bis] - Eastlake, D., Banerjee, A., Dutt, D., Perlman, R., and A. Ghanwani, "TRILL Use of IS-IS", draft-ietf-isis-rfc6326bis-00.txt, work in progress.

- [RFCclear] - Eastlake, D., M. Zhang, A. Ghanwani, V. Manral, A. Banerjee, draft-ietf-trill-clear-correct-06.txt, in RFC Editor's queue.
- [Channel] - D. Eastlake, V. Manral, Y. Li, S. Aldrin, D. Ward, "TRILL: RBridge Channel Support", draft-ietf-trill-rbridge-channel-08.txt, in RFC Editor's queue.
- [RFCfgl] - D. Eastlake, M. Zhang, P. Agarwal, R. Perlman, D. Dutt, "TRILL: Fine-Grained Labeling", draft-ietf-trill-fine-labeling-05.txt, work in progress.
- [ESADI] - Zhai, H., F. Hu, R. Perlman, D. Eastlake, J. Hudson, "TRILL (Transparent Interconnection of Lots of Links): The ESADI (End Station Address Distribution Information) Protocol", draft-ietf-trill-esadi-02.txt, work in progress.

9.2 Informational References

- [RFC5342] - Eastlake 3rd, D., "IANA Considerations and IETF Protocol Usage for IEEE 802 Parameters", BCP 141, RFC 5342, September 2008
- [DirectoryFramework] - Dunbar, L., D. Eastlake, R. Perlman, I. Gashinsky, "TRILL Edge Directory Assistance Framework", draft-ietf-trill-directory-framework-03.txt, 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: (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
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

Yizhou Li
Huawei Technologies
101 Software Avenue,
Nanjing 210012 China

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

Copyright, Disclaimer, and Additional IPR Provisions

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License. The definitive version of an IETF Document is that published by, or under the auspices of, the IETF. Versions of IETF Documents that are published by third parties, including those that are translated into other languages, should not be considered to be definitive versions of IETF Documents. The definitive version of these Legal Provisions is that published by, or under the auspices of, the IETF. Versions of these Legal Provisions that are published by third parties, including those that are translated into other languages, should not be considered to be definitive versions of these Legal Provisions. For the avoidance of doubt, each Contributor to the IETF Standards Process licenses each Contribution that he or she makes as part of the IETF Standards Process to the IETF Trust pursuant to the provisions of RFC 5378. No language to the contrary, or terms, conditions or rights that differ from or are inconsistent with the rights and licenses granted under RFC 5378, shall have any effect and shall be null and void, whether published or posted by such Contributor, or included with or in such Contribution.

TRILL Working Group
INTERNET-DRAFT
Intended status: Proposed Standard

Donald Eastlake
Yizhou Li
Weiguo Hao
Huawei
Ayan Banerjee
Insieme
February 16, 2013

Expires: August 15, 2013

TRILL: Vendor Specific TRILL Channel Protocol
<draft-eastlake-trill-vendor-channel-00.txt>

Abstract

The IETF TRILL (TRansparent Interconnection of Lots of Links) protocol is implemented by devices called TRILL switches or RBridges (Routing Bridges). TRILL includes a general mechanism, called RBridge Channel, for the transmission of typed messages between RBridges in the same campus and between RBridges and end stations on the same link. This document specifies how to send vendor specific messages over the RBridge Channel facility.

Status of This Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

Distribution of this document is unlimited. Comments should be sent to the TRILL working group mailing list.

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/lid-abstracts.html>. The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

Table of Contents

1. Introduction.....	3
1.1 Terminology and Acronyms.....	3
2. Vendor Channel Packet Format.....	4
3. Vendor Channel Errors.....	6
3.1 Sending an Error Response.....	6
4. IANA Considerations.....	8
5. Security Considerations.....	9
Normative References.....	10
Informative References.....	10
Acknowledgements.....	10
Authors' Addresses.....	11

1. Introduction

The IETF TRILL (TRansparent Interconnection of Lots of Links) protocol [RFC6325] is implemented by devices called TRILL switches or RBridges. It provides efficient least cost transparent frame routing in multi-hop networks with arbitrary topologies and link technologies, using link-state routing and a hop count. Links between TRILL switches can be arbitrary technology and, in general, the TRILL way to address or specify a TRILL switch (RBridge) in the interior of a TRILL campus is by its TRILL provided nickname [RFC6325] [ClearCorrect].

The TRILL protocol includes an RBridge Channel facility [RFCchannel] to support typed message transmission between RBridges in the same campus and between RBridges and end stations on the same link. This document specifies a method of sending messages specific to a particular organization, indicated by OUI (Organizationally Unique Identifier [RFC5342]), over the RBridge Channel facility.

Such organization specific messages can be used for vendor specific diagnostic or experimental messages.

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 the acronyms defined in [RFC6325] supplemented by the following additional acronym:

OUI - Organizationally Unique Identifier [RFC5342]

TRILL switch - An alternative term for an RBridge

2. Vendor Channel Packet Format

The general structure of an RBridge Channel packet on a link between RBridges (TRILL switches) is shown in Figure 1 below. When an RBridge Channel message is sent between an RBridge and an end station on the same link, in either direction, the TRILL Header is omitted. The type of RBridge Channel packet is given by a Protocol field in the RBridge Channel Header which indicates how to interpret the Channel Protocol Specific Payload. See [RFCchannel].

Frame Structure

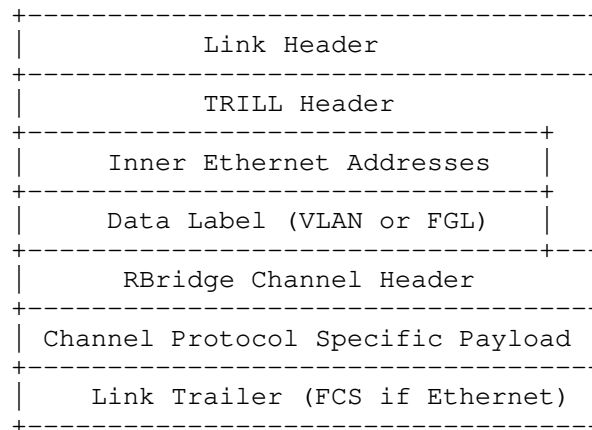


Figure 1. RBridge Channel Packet Structure

Figure 2 below expands the RBridge Channel Header and Channel Protocol Specific Payload above for the case of the Vendor Specific RBridge Channel Tunnel Protocol.

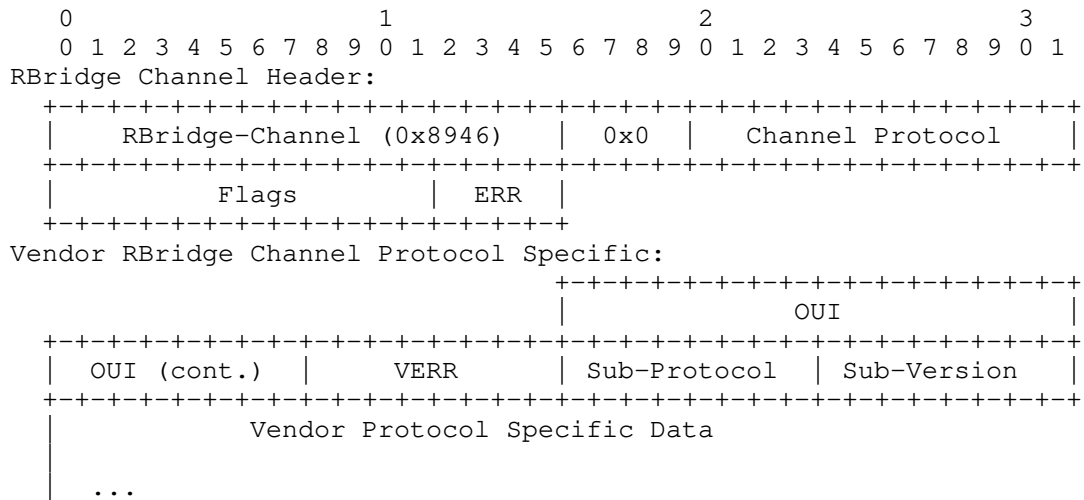


Figure 2. Channel Tunnel Message Structure

The fields in Figure 2 related to the Vendor RBridge Channel Protocol are as follows:

Channel Protocol: The RBridge Channel Protocol value allocated for Vendor Channel (see Section 4).

OUI: The field indicates the vendor specifying the particular use or uses of the Vendor Channel. The vendor to whom the OUI in this field has been allocated is in charge of specifying Vendor Channel messages using their OUI.

VERR: Vendor Channel Error. See Section 3.

Sub-Protocol: Actually, the vendor specifying their use of the Vendor Channel can do whatever they want with the bits after the VERR field. But it is strongly recommended that they use the sub-protocol / sub-version fields so that multiple and evolving uses can be specified based on a single OUI.

Sub-Version: See explanation above of the Sub-Protocol field. This field is provided to indicate the version of the particular vendor's Sub-Protocol.

3. Vendor Channel Errors

The VERR field values from 0x0 through 0xF inclusive are reserved for specification by the IETF. See Section 4. All other non-zero values of VERR are available for whatever use the vendor specifies except that a Vendor Channel implementation MUST NOT send a Vendor Channel Error in response to a Vendor Channel message with a non-zero VERR field.

The IETF specified VERR values thus far are as follows:

0. The VERR field is zero in Vendor Channel messages unless the the Vendor Channel packet is reporting an error.
1. The value one indicate that the OUI field value is unknown. If an RBridge implements the Vendor Channel facility and receives a Vendor Channel packet with a zero VERR field and an OUI field it does not recognize and the SL flag is zero in the RBridge Channel Header, it MUST set the VERR field to the value one and returns the packet as described in Section 3.1.
2. The value two indicates that the Sub-Protocol field value is unknown. If an RBridge implements the Vendor Channel facility and receives a Vendor Channel packet with a zero VERR field and zero SL flag in the RBridge Channel Header, an OUI that it implements, but a Sub-Protocol fields value it does not recongize, it SHOULD set the VERR field to the value two and returns the packet as described in Section 3.1.
3. The value three indicates that the Sub-Version field value is unknown. If an RBridge implements the Vendor RBridge Channel facility and receives a Vendor Channel packet with a zero VERR field and zero SL flag in the RBridge Channel Header, an OUI and Sub-Protocol that it implements, but a Sub-Version fields value it does not recongize, it SHOULD set the VERR field to the value three and returns the packet as described in Section 3.1.

3.1 Sending an Error Response

The IETF specified Vendor Channel error response are sent in response to a received RBridge Channel packet by setting the VERR field as specified above and modifying the packet as specified below.

The RBridge Channel Header is modified by setting the SL flag. (The ERR field will be zero because, if it was non-zero, the packet would have been handled at the RBridge Channel rather than being passed down to the Vendor Channel level.)

- o If Vendor Channel message was sent between RBridges, the TRILL Header is modified by clearing the M bit, setting the egress nickname to the ingress nickname as received, and setting the ingress nickname to a nickname held by the TRILL switch sending the error packet.
- o If Vendor Channel message was sent between an RBridge and an end station in either direction, the outer MAC addresses are modified by setting the Outer.MacDA to the Outer.MacSA as received, and the Outer.MacSA is set to the MAC address of the port of the TRILL switch or end station sending the error packet.
- o The priority of the error response message MAY be reduced from the priority of the Vendor Channel message causing the error, unless it was already minimum priority, and MAY set the Drop Eligibility Indicator bit in an error response. (Priorities are ordered from highest to lowest as 7, 6, 5, 4, 3, 2, 0, and 1. See Section 4.1.1, [RFC6325].)

It is generally anticipated that the entire packet in which an error was detected would be sent back, modified as above, so that, for example, error responses could more easily be matched with messages sent; however, this is really up to the vendor specifying how their Vendor RBridge Channel messages are to be used.

4. IANA Considerations

IANA is requested to allocate TBD for Vendor Specific RBridge Channel Protocol from the range of RBridge Channel protocols allocated by Standards Action.

IANA is requested to establish a "Vendor RBridge Channel Error Codes" registry with initial entries as follows:

Code	Description	Reference
----	-----	-----
0	No error	This document
1	Unknown OUI	This document
2	Unknown Sub-Protocol	This document
3	Unknown Sub-Version	This document
0x04-0x0F	Allocated by Standards Action	-
0x10-0xFF	Reserved for vendor use	This document

5. Security Considerations

See [RFC6325] for general TRILL Security Considerations.

See [RFCchannel] for general RBridge Channel Security Considerations.

The Vendor Specific RBridge Channel Protocol provides no security assurances or features. Any needed security could be provided by fields or processing within the Vendor Protocol Specific Data, which is outside the scope of this document. Alternatively, use of Vendor Channel could be nested inside the RBridge Channel Tunnel Protocol [RFCtunnel] which can provide some security services.

Normative References

- [RFC2119] - Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC5342] - Eastlake 3rd, D., "IANA Considerations and IETF Protocol Usage for IEEE 802 Parameters", BCP 141, RFC 5342, September 2008.
- [RFC6325] - Perlman, R., D. Eastlake, D. Dutt, S. Gai, and A. Ghanwani, "RBridges: Base Protocol Specification", RFC 6325, July 2011.
- [RFCchannel] - D. Eastlake, V. Manral, L. Yizhou, S. Aldrin, D. Ward, "TRILL: RBridge Channel Support", draft-ietf-trill-rbridge-channel-08.txt, in RFC Editor's queue.
- [ClearCorrect] - Eastlake, D., M. Zhang, A. Ghanwani, V. Manral, A. Banerjee, "TRILL: Clarifications, Corrections, and Updates", draft-ietf-trill-clear-correct, work in progress.

Informative References

- [RFCtunnel] - Eastlake, D., ... "TRILL: Channel Tunnel", draft-eastlake-trill-channel-tunnel, work in progress.

Acknowledgements

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

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

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

Weiguo Hao
Huawei Technologies
101 Software Avenue,
Nanjing 210012, China

Phone: +86-25-56623144
Email: haoweiguo@huawei.com

Ayan Banerjee
Insieme Networks
210 West Tasman Drive
San Jose, CA 95134 USA

Email: ayabaner@gmail.com

Copyright, Disclaimer, and Additional IPR Provisions

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License. The definitive version of an IETF Document is that published by, or under the auspices of, the IETF. Versions of IETF Documents that are published by third parties, including those that are translated into other languages, should not be considered to be definitive versions of IETF Documents. The definitive version of these Legal Provisions is that published by, or under the auspices of, the IETF. Versions of these Legal Provisions that are published by third parties, including those that are translated into other languages, should not be considered to be definitive versions of these Legal Provisions. For the avoidance of doubt, each Contributor to the IETF Standards Process licenses each Contribution that he or she makes as part of the IETF Standards Process to the IETF Trust pursuant to the provisions of RFC 5378. No language to the contrary, or terms, conditions or rights that differ from or are inconsistent with the rights and licenses granted under RFC 5378, shall have any effect and shall be null and void, whether published or posted by such Contributor, or included with or in such Contribution.

TRILL

Internet Draft

Intended status: Standards Track

Expires: July 2013

Weiguo Hao
Yizhou Li
Huawei Technologies
January 16, 2013

The problem statement of RBridge edge group state synchronization
draft-hao-trill-rb-syn-00.txt

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79. This document may not be modified, and derivative works of it may not be created, and it may not be published except as an Internet-Draft.

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79. This document may not be modified, and derivative works of it may not be created, except to publish it as an RFC and to translate it into languages other than English.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

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

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

This Internet-Draft will expire on July 16, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

Abstract

In TRILL multi-homing scenario, the concept of virtual RBridge in [TRILLPN], was introduced to address the MAC flip-flopping problem at remote RBridges. Based on virtual RBridge mechanism, Coordinated Multicast Trees (CMT) solution in [CMT] was introduced to solve the related RPF issues. In this document, additional problems are described regarding virtual Bridges members' state synchronization in multi-homing scenario, including virtual RBridge membership auto discovery, pseudo-nickname static configuration consistency check, dynamic pseudo-nickname allocation, CMT configuration synchronization, LACP configuration and state synchronization, and node/link failure detection. To address these problems, a communication protocol among members of a virtual RBridge group should be provided. Requirements for this protocol is also discussed.

Table of Contents

1. Introduction	3
2. Conventions used in this document.....	5
3. Problem Statement	6
3.1. RBv membership configuration and state synchronization..	6
3.2. CMT configuration and state synchronization	7
3.3. LACP configuration and state synchronization	8
4. Requirements for communication protocol in RBv	10
5. Security Considerations.....	12
6. IANA Considerations	12
7. References	12
7.1. Normative References.....	12
7.2. Informative References.....	13
8. Acknowledgments	14

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) as its control plane and encapsulates native frames with a TRILL header.

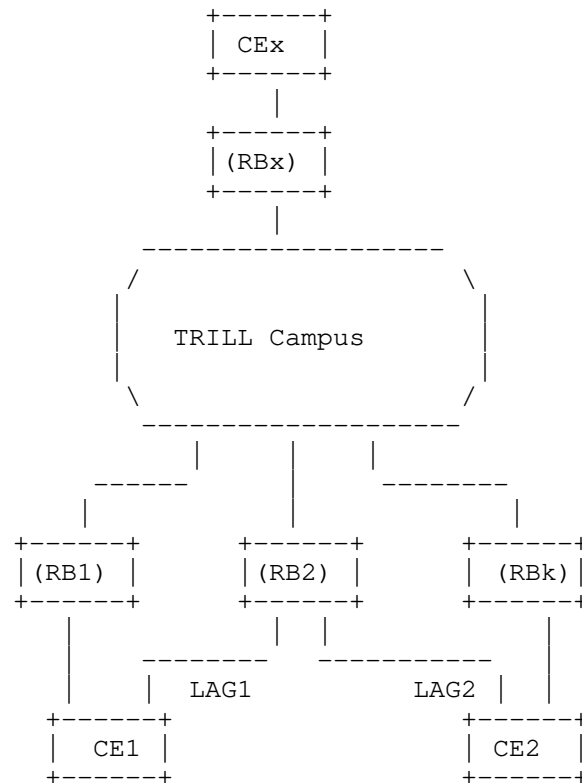


Figure 1 Reference Topology

In order to improve the reliability of connection to TRILL network , CE devices typically are multi-homed to edge RBridges and treat all of the uplinks as a single Link Aggregation (LAG) bundle [802.1AX] in the scenario shown by Figure 1. In this scenario, When remote RBridge RBx receives a frame originated by CE1, the ingress RBridge maybe either one of the edge RBridges i.e. RB1 or RB2. The learning on RBx for source MAC will flip-flop between RB1's and RB2's nicknames. In [TRILLPN], the concept of Virtual RBridge, along with

its pseudo-nickname, is introduced to address the MAC flip-flopping problem in remote RBridges.

A Virtual RBridge (RBv) represents a group of different ports on different edge RBridges, on which these RBridges provide end-station service to a set of their attached CE devices. After joining RBv, such an RBridge port is called a member port of RBv, and such an RBridge becomes a member RBridge of RBv. In an RBridge RBv is identified by its virtual nickname in TRILL campus, and virtual nickname is also referred to as pseudo-nickname in this specification.

An RBridge port can join at most one RBv at any time, but different ports on the same RBridge can join the same RBv or different RBvs. After joining an RBv, such a port becomes a member port of the RBv, and the RBridge becomes a member RBridge of the RBv.

Furthermore, for a member RBridge, it MUST move out of RBv and clear the RBv's information from its self-originated LSPs when it loses the last member port from this group, due to port down, configuration, and etc.

Based on the concept of Virtual RBridge and pseudo-nickname, Coordinated Multicast Trees (CMT) [CMT] solution was introduced to solve the related RPF issues. In CMT solution, different member RBridges are assigned different distribution trees for forwarding the multi-destination TRILL data frames that using RBv's pseudo-nickname as ingress nickname in their TRILL header.

When a member RBridge joins into or leaves from a virtual RBridge group RBv due to its last member ports up/down or its configuration changing, the distribution trees assigned to different member RBridges may change.

For TRILL multi-homing scenario, pseudo-nickname and CMT is not sufficient to provide a complete solution. Additional problems such as RBv membership management, LACP configuration and state synchronization, node and access link failure detection, and etc still exist. This draft is going to talk about those problem in more details.

2. Conventions used in this document

In examples, "C:" and "S:" indicate lines sent by the client and server respectively.

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.

In this document, the characters ">>" preceding an indented line(s) indicates a compliance requirement statement using the key words listed above. This convention aids reviewers in quickly identifying or finding the explicit compliance requirements of this RFC.

TRILL: Transparent Interconnection of Lots of Links. TRILL 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) as its control plane and encapsulates native frames with a TRILL header.

RB: Router Bridge. RBs are a switch that implement implement the TRILL protocol and combine the advantages of bridges and routers.

CMT: Coordinated Multicast Trees.

LAG: Link Aggregation, as specified in [8021AX].

LACP: Link Aggregation Control Protocol.

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

3. Problem Statement

For TRILL multi-homing scenario, the following problems should be addressed:

3.1. RBv membership configuration and state synchronization

A Virtual RBridge (RBv) is identified by its virtual nickname referred as pseudo-nickname in [PSEUDO-NICK]. RBv must allow static member configuration by network operator.

If each member of RBv statically configures its RBridge ports with a pseudo-nickname, the pseudo-nickname should be consistent among all member RBridges in RBv. Communication protocol between member

RBridges should be provided to ensure pseudo-nickname configuration consistency in RBv. Member RBridges in RBv should notify each other to find if conflict of pseudo-nickname configuration exists when pseudo-nickname is configured. If conflict exists, It is recommended to send trap to network management system (NMS) and let operator modify configuration to eliminate conflict. Only when the conflict is removed, each member RBridge can advertises the RBv's pseudo-nickname using the nickname sub-TLV [rfc6326bis], along with its regular nickname(s), in its LSPs.

The communication protocol is an inter-chassis communication protocol among RBridges in RBv to synchronize configuration and/or running state data. The communication protocol should run over TRILL campus to accommodate multi-hop interconnection among member RBridges in RBv.

To simplify configuration of pseudo-nickname, dynamic pseudo-nickname allocation through communication protocol should be allowed. For TRILL VLAN-x Appointed Forwarder, TRILL Hello protocol specified in [RFC6325] is used for DRB election and for VLAN-x AF's appointment on those ports. Pseudo-nickname can be dynamic allocated by DRB and be notified to other member RBs through TRILL Hello. For LAG multi-homed access scenario, as there is no HELLOs on LAG RBv membership auto-discovery and pseudo-nickname dynamic allocation are not achievable using the Hello based mechanism. Some new method is required for such purpose. One of the potential ways is to use the member communication protocol as follows.

As all member RBridges in RBv can exchange message through TRILL campus although there is no HELLOs on LAG access port side, dynamic pseudo-nickname allocation can be accomplished through communication protocol over TRILL campus. The member RBridges in RBv select one RB as DRB and let DRB assign pseudo-nickname dynamically. After pseudo-nickname is allocated, each member RBridge in RBv can advertises the RBv's pseudo-nickname in its LSPs.

3.2. CMT configuration and state synchronization

CMT configuration should be synchronized between RBridges in RBv to ensure different member RBridges assigned to different distribution trees. If different RBridges in one RBv associate the same virtual RBridge as their child in the same tree or trees, conflict occurs and there should be a mechanism to remove the conflict. It is recommended to send trap to NMS if conflict occurs. Network operator may manually eliminate the conflict by modify configurations.

Automatic mechanism should also be provided to remove the conflict. After the conflict is removed in local RBv, RBridges can advertise Affinity sub-TLVs to trill campus.

If RBv membership changes when a member RBridges joins or leaves RBv, each member RBridge in the RBv should do configuration consistency check first. If no conflict is found or the conflict had been removed, each member RBridge in the RBv recalculates the multi-destination tree assignment and advertises the related trees using Affinity sub-TLV.

For member RBridges node and link (all member link of LAG) failure, other RBridges in the RBv should detect as soon as possible to achieve fast failure recovery. Upon member RBridges node and link (all member link of LAG) failure detection, other member RBridges in the RBv will recalculate the multi-destination tree assignment and advertise the related trees using Affinity sub-TLV.

So for CMT, communication protocol between member RBridges also should be provided to achieve CMT configuration synchronization, conflict elimination, node and link failure detection, and RBv membership auto-discovery.

3.3. LACP configuration and state synchronization

In IEEE802.1AX standard The Link Aggregation Control Protocol (LACP) provides a standardized means for exchanging information between Partner Systems on a link to allow their Link Aggregation Control instances to reach agreement on the identity of the Link Aggregation Group to which the link belongs, move the link to that Link Aggregation Group, and enable its transmission and reception functions in an orderly manner. The aggregated ports in one LAG are located on one switch and can't be located on two different switches or chassis' in different locations. since IEEE802.1AX?Link Aggregation is only defined for a single system, the redundancy is limited to a point to point connection between two devices and a complete system failure on one end will bring down the LAG.

In the scenario that CE multi-homing to multiple RBridges in a edge group link aggregation groups spanning two or multiple systems should be provided. The standard as defined in IEEE802.1AX doesn't provide for this. To support CE multi-homing with multi-chassis Ethernet bundles, [802.1AX] LACP state should be synchronized or shared between these systems. This ensures that the RBs can present a single LACP bundle to the CE. This is required for initial system bring-up and upon any configuration change.

Just similar to the description in [EVPN], at least the following LACP specific configuration parameters should be synchronized amongst RBs in RBv:

- System Identifier (MAC Address): uniquely identifies a LACP speaker.
- System Priority: determines which LACP speaker's port priorities are used in the Selection logic.
- Aggregator Identifier: uniquely identifies a bundle within a LACP speaker.
- Aggregator MAC Address: identifies the MAC address of the bundle.
- Aggregator Key: used to determine which ports can join an Aggregator.
- Port Number: uniquely identifies an interface within a LACP speaker.
- Port Key: determines the set of ports that can be bundled.
- Port Priority: determines a port's precedence level to join a bundle in case the number of eligible ports exceeds the maximum number of links allowed in a bundle.

Furthermore, the RBs should also synchronize operational (run-time) data, in order for the LACP Selection logic state-machines to execute. This operational data includes the following LACP operational parameters, on a per port basis:

- Partner System Identifier: this is the CE System MAC address.
- Partner System Priority: the CE LACP System Priority

- Partner Port Number: CE's AC port number.
- Partner Port Priority: CE's AC Port Priority.
- Partner Key: CE's key for this AC.
- Partner State: CE's LACP State for the AC.
- Actor State: RB's LACP State for the AC.
- Port State: RB's AC port status.

The operational state needs to be communicated between RBs forming a multi-chassis bundle during LACP initial bringup, upon any configuration change and upon the occurrence of a failure.

If member RBridge of the virtual RBridge group has any node failure, other RBridges of the group should invoke the Selection Logic and select new SELECTED port. The failure detection timer is critical to failure recovery performance. It is desired to achieve sub-second detection of node failure (~ 50 - 150 msec) in order to ensure application SLA(service level agreement).

Upon detection of local link failure, RB1 in the RBv should notify other RBs in the RBv immediately. Then other RBs in the RBv should invoke the Selection Logic and select new SELECTED port as well. Immediate notification of access-link state(up/down etc) changes should also be provided to accomplish fast failure recovery. In other words, the transmission of messages carrying link state of the LAG should be on-demand rather than timer-based to minimize inter-chassis state synchronization delay.

4. Requirements for communication protocol in RBv

In summary, a communication protocol between member RBridges in RBv should be provided to accomplish multi-homing access model. The communication protocol is restricted to RBridge nodes in RBv edge group and is used for configuration and state synchronization. It is expected that LSP would not be used for this purpose since it may cause campus wide fluctuation. Local behavior is preferred. After member RBridges in RBv discover each other and establish connection between each other, they can proceed with further state and configuration synchronization which are addressed in the following point.

The communication should accommodate multi-hop interconnection between RBridges over TRILL campus. Because RBridges in RBv can't

exchange information over access link of LAG, so RBridges in RBv should exchange information over TRILL campus. The suggested control channel for communication between member RBridges in RBv to exchange state and configuration information is RBridge channel. Each member RBridge establish connection to other RBridges of same RBv over RBridge channel. This assumes that resiliency mechanisms are in place to protect the route to the remote RBridge nodes, and hence loss of TRILL data layer reachability to a given node can only mean that the node itself has failed.

The communication protocol should satisfy the following requirements:

1. Support RBv membership static configuration and auto-discovery. A mechanism that enables RB nodes to manage their RBv Membership should be defined. RBv membership auto-discovery can simplify configuration of RBv. After member RBridges in RBv discover each other and establish connection between each other, the state and configuration can be synchronized among them which are discussed in the following point.
2. Support consistency check for static pseudo-nickname configuration consistency. The pseudo-nickname configured on each member RBridges in RBv should be same. If conflict exists, It is recommended to send trap to NMS and let operator modify configuration to eliminate conflict. Only when the conflict is removed, each member RBridge in RBv can advertises the RBv's pseudo-nickname in its LSPs.
3. Support dynamic pseudo-nickname allocation. To simplify configuration of pseudo-nickname, dynamic pseudo-nickname allocation through communication protocol should be allowed. After pseudo-nickname is allocated, each member RBridge in RBv can advertises the RBv's pseudo-nickname in its LSPs.
4. Support CMT configuration synchronization and conflict elimination. CMT configuration should be synchronized in RBv to ensure different member RBridges are assigned different distribution trees. If conflict occurs, i.e. one tree is used by more than one members, It is recommended to send trap to NMS. Conflict elimination can rely on operator or automatic mechanism. After the conflict is removed in local RBv, RBridges advertise Affinity sub-TLVs to trill campus.
5. Support fast node failure detection. Upon detection other member RBridges node failure, RBridges in RBv should invoke LACP re-selection Logic and CMT re-calculation algorithm.

The communication protocol can either define its own keepAlive mechanism for purpose of node failure detection or reuse existing fault detection mechanisms. BFD over TRILL and TRILL OAM for RB reachability monitoring are existing fault detection mechanisms and may be used to detect RBridges node failure.

6. Support fast link failure detection. When a member RBridge in RBv detects a failure of its access link, it should send an link failure notification message immediately to inform other member RBridges. Other member RBridges in RBv should invoke LACP re-selection Logic and CMT re-calculation algorithm similar to node failure process.
7. Support LACP configuration and state synchronization. To support CE multi-homing with multi-chassis Ethernet bundles, LACP state should be synchronized or shared between these systems. For CE device, all RBridges in virtual RBridge group simulate one LACP end system and perform same LACP selection logic. Member RBridges in RBv can use RBridge channel as control channel to exchange LACP configuration and state synchronization between each other.

Additional requirements considerations such as flow-control, reliable and in-order message delivery, and etc are being discussed.

5. Security Considerations

This document does not change the general TRILL security considerations of the TRILL base protocol.

In the scenario where the members of an RBv are located in different physical locations and connected over TRILL campus, transport security between devices in an RBv should be provided with secure authentication mechanism built into the communication protocol.

6. IANA Considerations

If RBridge channel is used for control channel of communication protocol in RBv, then IANA is requested to allocate the new RBridge channel protocol codes.

7. References

7.1. Normative References

[RFC6325] Perlman, R., Eastlake 3rd, D., Dutt, D., Gai, S., and A.

Ghanwani, "Routing Bridges (RBridges): Base Protocol Specification", RFC 6325, July 2011.

[RFC6326] Eastlake, D., Banerjee, A., Dutt, D., Perlman, R., and A. Ghanwani, "TRILL Use of IS-IS", RFC 6326, July 2011.

[6326bis] Eastlake, D. et.al., " Transparent Interconnection of Lots of Links (TRILL) Use of IS-IS", draft-eastlake-isisrfc6326bis-07.txt, Work in Progress, December 2011.

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

[TRILLChannel] - Eastlake, D., V. Manral, Y. Li, S. Aldrin, D. Ward, "RBridges: RBridge Channel Support in TRILL", draft-ietftrill-rbridge-channel, work in progress.

[RFC6327] Eastlake 3rd, D., Perlman, R., Ghanwani, A., Dutt, D., and V. Manral, "Routing Bridges (RBridges): Adjacency", RFC 6327, July 2011

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

[TRILL-CMT] " Coordinated Multicast Trees (CMT) for TRILL ", draft-ietf-trill-cmt-01, November 2012.

[8021AX] IEEE, " Link Aggregation ", 802.1AX-2008, 2008.

[EVPN] " BGP MPLS Based Ethernet VPN ", draft-ietf-l2vpn-evpn-02, October 2012.

7.2. Informative References

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

[802.1D] "IEEE Standard for Local and metropolitan area networks
/Media Access Control (MAC) Bridges", 802.1D-2004, 9 June
2004.

8. Acknowledgments

The authors wish to acknowledge the important contributions of
Changbao Liu, Donald Eastlake, Mingui Zhang.

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

INTERNET-DRAFT
Intended Status: Informational

Linda Dunbar
Donald Eastlake
Huawei
Radia Perlman
Intel
Igor Gashinsky
Yahoo
February 23, 2013

Expires: August 22, 2012

TRILL (Transparent Interconnection of Lots of Links):
Edge Directory Assistance Framework
<draft-ietf-trill-directory-framework-04.txt>

Abstract

Edge RBridges currently learn the mapping between MAC addresses and their egress RBridges by observing the data packets they ingress or egress or by the TRILL ESADI protocol. When an ingress RBridge receives a data frame whose destination address (MAC&VLAN) that RBridge does not know, the data frame is flooded within the VLAN across the TRILL campus.

This document describes the framework for using directory services to assist edge RBridges in reducing multi-destination frames, particularly unknown unicast frames flooding, and ARP/ND, thus improving TRILL (Transparent Interconnection of Lots of Links) network scalability.

Status of This Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

Distribution of this document is unlimited. Comments should be sent to the TRILL working group mailing list.

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at
<http://www.ietf.org/lid-abstracts.html>. The list of Internet-Draft
Shadow Directories can be accessed at
<http://www.ietf.org/shadow.html>.

Table of Contents

1. Introduction.....	4
2. Terminology.....	5
3. Impact of Massive Number of End Stations.....	6
3.1 Issues of Flooding Based Learning in Data Centers.....	6
3.2 Two Examples.....	7
4. Benefits of Directory Assisted Edge RBridge.....	8
5. Generic operation of Directory Assistance.....	10
5.1 Information in Directory for Edge RBridges.....	10
5.2 Push Model and Requirements.....	10
5.3 Pull Model and Requirements.....	12
6. Recommendation.....	14
7. Security Considerations.....	14
8. IANA Considerations.....	14
9. Acknowledgements.....	14
10. References.....	15
10.1 Normative References.....	15
10.2 Informative References.....	15
Authors' Addresses.....	16

1. Introduction

Edge RBridges (devices implementing [RFC6325], also known as TRILL Switches) currently learn the mapping between destination MAC addresses and their egress RBridges by observing data packets or by the ESADI (End Station Address Distribution Information) protocol. When an ingress RBridge receives a data frame for a destination address (MAC&VLAN) that RBridge does not know, the data frame is flooded within that VLAN across the TRILL campus.

This document describes a framework for using directory services to assist edge RBridges by reducing multi-destination frames, particularly ARP [RFC826], ND [RFC4861], and unknown unicast, improving TRILL network scalability in environments where a directory can be available, such as data centers.

Data center networks differ from enterprise campus networks in several ways that make them attractive for the use of directory assistance, in particular:

1. Data centers, especially Internet and/or multi-tenant data centers tend to have a large number of end stations with a wide variety of applications.
2. Topology is often based on racks and rows. Furthermore, guest operating system assignment to Servers, Racks, and Rows is orchestrated by a Server/VM (virtual machine) Management system, not done at random. So the information necessary for a directory is normally available.
3. Rapid workload shifting in data centers can accelerate the frequency of the physical servers being re-loaded with different applications. Sometimes, the applications loaded into one physical server at different times can belong to different subnets. When a VM is moved to a new location or a server is loaded with a new application with different IP/MAC addresses, it is more likely that the destination address of data packets sent out from those VMs are unknown to their attached edge RBridges.
4. With server virtualization, there is an increasing trend to dynamically create or delete VMs when demand for resource changes, to move VMs from overloaded servers to less loaded servers, or to aggregate VMs onto fewer servers when demand is light. This results in the more common occurrence of multiple subnets on the same port at the same time and a higher change rate for VMs than for physical servers.

Both items 3 and 4 above can lead to applications in one subnet being placed in different locations (racks or rows) or one rack having applications belonging to different subnets.

2. Terminology

The terms "Subnet" and "VLAN" are used interchangeably in this document because it is common to map one subnet to one VLAN.

Bridge: IEEE Std 802.1Q-2011 compliant device [802.1Q]. In this document, Bridge is used interchangeably with Layer 2 switch.

EoR: End of Row switches in data center. Also known as aggregation switches.

End Station: Guest OS running on a physical server or on a virtual machine. An end station in this document has at least one IP address and at least one MAC address.

IS-IS: Intermediate System to Intermediate System. TRILL uses IS-IS [IS-IS] [RFC6326].

RBridge: "Routing Bridge", an alternative name for a TRILL switch.

Station: A node, or a virtual node, with IP and/or MAC addresses.

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

TRILL: Transparent Interconnection of Lots of Links [RFC6325]

TRILL switch: A device implementing the TRILL protocol [RFC6325]

VM: Virtual Machine

3. Impact of Massive Number of End Stations

This section discusses the impact of a massive number of end stations in a TRILL campus using Data Centers as an example.

3.1 Issues of Flooding Based Learning in Data Centers

It is common for Data Center networks to have multiple tiers of switches, for example, one or two Access Switches for each server rack (ToR), aggregation switches for some rows (or EoR switches), and some core switches to interconnect the aggregation switches. Many aggregation switches deployed in data centers have high port density. It is not uncommon to see aggregation switches interconnecting hundreds of ToR switches.

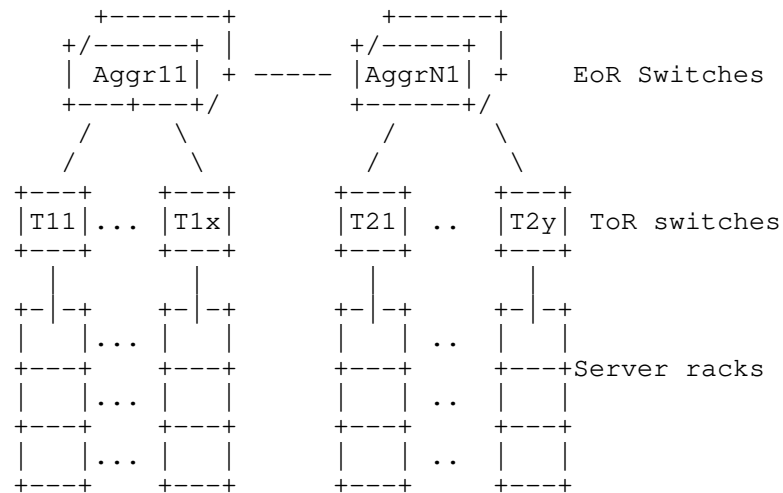


Figure 1: Typical Data Center Network Design

The following problems could occur when TRILL is deployed in a data center with large number of end stations and the end stations in one subnet/VLAN could be placed under multiple edge RBridges:

- Unnecessary filling of slots in the MAC address learning table of edge RBridges, e.g. RBridge T11, due to T11 receiving broadcast / multicast traffic (e.g. ARP/ND, cluster multicast, etc.) from end stations under other edge RBridges that are not actually communicating with any end stations attached to T11.
- Packets being flooded across TRILL campus when their destination MAC addresses are not in ingress RBridge's MAC address to egress RBridge cache.

- In an environment where VMs migrates, there is higher chance of cached information becoming invalid, causing traffic to be black-holed by the ingress RBridge, that is, persistently sent to the wrong egress RBridge. If VMs do not flood gratuitous ARP/ND or VDP [802.1Qbg] messages upon arriving at new locations, the ingress nodes might not have MAC entries for the MAC of the newly arrived VMs, causing unknown address flooding.

3.2 Two Examples

Consider a data center with 1600 server racks. Each server rack has at least one ToR switch. The ToR switches are further divided into 8 groups, with each group being connected by a set of aggregation switches. There could be 4 to 8 aggregation switches in each set to achieve load sharing for traffic to/from server racks. If TRILL is deployed in this data center environment, let's consider the following two scenarios for the TRILL campus boundary:

- Scenario #1: TRILL campus boundary starts at ToR switches:

If each server rack has one ToR, there are 1600 edge RBridges. If each rack has two ToR switches, then there will be 3200 edge RBridges

In this scenario, the TRILL domain will have more than 1600 (or 3200) + 8*4 (or 8*8) nodes, which is a large IS-IS domain. Even though a mesh IS-IS domain can scale up to thousands of nodes, it is challenging for aggregation switches to handle IS-IS link state advertisement among hundreds of parallel ports.

- Scenario #2: TRILL campus boundary starts at the aggregation switches:

With the same assumptions as before, the number of nodes in the TRILL campus will be less than 100, and aggregation switches don't have to handle IS-IS link state advisements among hundreds of parallel ports.

However, the number of MAC&VLAN<->Egress RBridge Mapping entries to be learned and managed by RBridge edge node can be very large. In the example above, each edge RBridge has 200 edge ports facing the ToR switches. If each ToR has 40 downstream ports facing servers and each server has 10 VMs, there could be $200*40*10 = 80000$ end stations attached. If all those end stations belong to 1600 VLANs (i.e. 50 per VLAN) and each VLAN has 200 end stations, then under the worst-case scenario, the total number of MAC&VLAN entries to be learned by the edge RBridge can be $1600*200=320000$, which is very large.

4. Benefits of Directory Assisted Edge RBridge

In some environments, particularly data centers, the assignment of applications to servers, including rack and row selection, is orchestrated by Server (or VM) Management System(s). That is, there is a database or multiple databases (distributed model) that have the knowledge of where each application is placed. If the application location information can be fed to RBridge edge nodes, in some form of Directory Service, then there is much less chance of RBridge edge nodes receiving unknown MAC destination address, therefore less chance of flooding.

Avoiding unknown unicast address flooding to the TRILL campus is especially valuable in the data center environment because there is a higher chance of an edge RBridge receiving packets with unknown unicast destination address and broadcast / multicast messages due to VM migration and servers being loaded with different applications. When a VM is moved to a new location or a server is loaded with a new application with a different IP/MAC addresses, it is more likely that the destination address of data packets sent out from those VMs are unknown to their attached edge RBridges. In addition, gratuitous ARP (IPv4, [RFC826]) or Unsolicited Neighbor Advertisement (IPv6, [RFC4861]) sent out from those newly migrated or activated VMs have to be flooded to other edge RBridges that have VMs in the same subnets.

The benefits of using directory assistance include:

- Avoid flooding unknown unicast destination address across TRILL campus. The Directory enforced MAC&VLAN <-> Egress RBridge mapping table can determine if a data packet needs to be forwarded across TRILL campus.

When multiple RBridge edge ports are connected via a bridged LAN to end stations (servers/VMs), a directory assisted edge RBridge won't need to flood unknown unicast destination data frames to all ports of the edge RBridges in the frame's VLAN when it ingresses a frame. It can depend on the directory to tell it where the destination is. When the directory doesn't have the needed information, the frames can be dropped or flooded depending on the policy configured.

- Reduce flooding of decapsulated Ethernet frames with unknown MAC destination address to a bridged LAN connected to RBridge edge ports.

When an RBridge receives a TRILL data packet whose destination Nickname matches with its own, the normal procedure is for the RBridge to decapsulate it and forward the decapsulated Ethernet frame to the directly attached bridged LAN. If the destination

MAC is unknown, the RBridge floods the decapsulated Ethernet frame out all ports in the fame's VLAN. With directory assistance, the egress RBridge can determine if the MAC destination address in a frame matches any end stations attached via the bridged LAN. Frames can be discarded if their destination addresses do not match.

- Reduce the amount of MAC&VLAN <-> Egress RBridge mapping maintained by edge RBridges. There is no need for an edge RBridge to keep MAC entries of remote end stations that don't communicate with the end stations locally attached.
- Eliminate ARP/ND being broadcasted or multi-casted through the TRILL core.

5. Generic operation of Directory Assistance

5.1 Information in Directory for Edge RBridges

To achieve the benefits of directory assistance for TRILL, the corresponding directory server entries will need, at a minimum, the following logical attributes:

```
[{IP, MAC/VLAN, {list of attached RBridge nicknames}, {list of
interested RBridges}]
```

The {list of attached RBridges} are the edge RBridges to which the host (or VM) specified by the [IP or MAC/VLAN] in the entry is attached. The {list of interested RBridges} are the remote RBridges that might have attached hosts to communicate with the host in this entry.

When a host has multiple IP addresses, there will be multiple entries.

The {list of interested RBridges} could get populated when an RBridge queries for information, or pushed down from management systems. The list is used to notify those RBridges when the host (specified by the IP/MAC/VLAN) in the entry connectivity to its attached RBridges changes. An explicit list in the directory is not needed as long as the interested RBridges can be determined.

There are two different models for Directory assistance to edge RBridges: Push Model and Pull Model.

5.2 Push Model and Requirements

Under this model, Directory Server(s) push down the MAC&VLAN <-> Egress RBridge mapping for all the end stations that might communicate with end stations attached to an RBridge edge node. If the packet's destination address can't be found in the MAC&VLAN<->Egress RBridge table, the ingress RBridge could be configured to:

```
    simply drop a data packet,
    flood it to TRILL campus, or
    start the pull process to get information from directory
    server(s)
```

It may not be necessary for every edge RBridge to get the entire mapping table for all the end stations in a campus. There are many

ways to narrow the full set down to a smaller set of remote end stations that communicate with end stations attached to an edge RBridge. A simple approach is to only pushing down the mapping for the VLANs that have active end stations under an edge RBridge. This approach can reduce the number of mapping entries being pushed down.

However, the Push Model usually will push down more entries of MAC&VLAN<->Egress RBridge mapping to edge RBridges than needed. Under the normal process of edge RBridge cache aging and unknown destination address flooding, rarely used mapping entries would have been removed. But it can be difficult for Directory Servers to predict the communication patterns among applications within one VLAN. Therefore, it is likely that the Directory Servers will push down all the MAC&VLAN entries if there are end stations in the VLAN being attached to the edge RBridge. This is a disadvantage of the Push Model compared with the Pull Model described below.

In the Push Model, it is necessary to have a way for an RBridge node to request directory server(s) to start pushing down the mapping entries. This method should at least include the VLANs enabled on the RBridge, so that directory server doesn't need to push down the entire mapping entries for all the end stations in the campus. An RBridge must be able to get mapping entries when it is initialized or restarted.

The Push Model's detailed method and any handshake mechanism between RBridge and Directory Server(s) is beyond the scope of this framework document.

When a directory server needs to push down a large number of entries to edge RBridges, efficient data organization should be considered. For example, with one edge RBridge Nickname being associated with all attached end stations' MAC addresses and VLANs as shown below:

Nickname1	VID-1	IP/MAC1, IP/MAC2, ,, IP/MACn
	VID-2	IP/MAC1, IP/MAC2, ,, IP/MACn
	IP/MAC1, IP/MAC2, ,, IP/MACn
Nickname2	VID-1	IP/MAC1, IP/MAC2, ,, IP/MACn
	VID-2	IP/MAC1, IP/MAC2, ,, IP/MACn
		IP/MAC1, IP/MAC2, ,, IP/MACn
-----		IP/MAC1, IP/MAC2, ,, IP/MACn

Table 1: Summarized table pushed down from directory

Whenever there is any change in MAC&VLAN <-> Egress RBridge mapping, that can be triggered by end stations being added, moved, or de-commissioned, an incremental update can be sent to the edge RBridges which are impacted by the change. Therefore, something like a sequence number has to be maintained by directory servers and RBridges. Detailed mechanisms will be specified in a separate document.

5.3 Pull Model and Requirements

Under this model, an RBridge pulls the MAC&VLAN<->Egress RBridge mapping entry from the directory server when its cache doesn't have the entry. There are several possibilities to trigger the pulling process:

- The RBridge edge node can send a pull request whenever it receives an unknown MAC destination, or
- The RBridge edge node can intercept all ARP/ND requests and forward them or appropriate requests to the Directory Server(s) that has the information on where the target stations are located.
- The Pull Directory response could indicate that the address being queried is unknown or that the requestor is administratively prohibited from getting an informative response.

By using a Pull Directory, the frame with unknown MAC destination address doesn't have to be flooded across TRILL domain and the ARP/ND requests don't have to be broadcast or multicast across the TRILL

domain.

The ingress RBridge can cache the response pulled down from the directory. The timer for cache should be short in an environment where VMs move frequently. The cache timer could be configured by management system or could be sent down along with the Pulled reply by the directory server(s). It is important that the cached information be kept consistent with the actual placement of addresses in the campus; therefore, there needs to be some mechanism by which RBridges that have pulled information that has not expired can be informed when that information changes or the like.

One advantage of the Pull Model is that edge RBridges can age out MAC&VLAN entries if they haven't been used for a certain configured period of time or a period of time provided by the Directory. Therefore, each edge RBridge will only keep the entries that are frequently used, so mapping table size will be smaller. Edge RBridges would query the Directory Server(s) for unknown MAC destination addresses in data frames or ARP/ND and cache the response. When end stations attached to remote edge RBridges rarely communicate with the locally attached end stations, the corresponding MAC&VLAN entries would be aged out from the RBridge's cache.

An RBridge waiting for response from Directory Servers upon receiving a data frame with an unknown destination address is similar to an L2/L3 boundary router waiting for ARP/ND response upon receiving an IP data packet whose destination IP is not in the router's IP/MAC cache table. Most deployed routers today do hold the packet and send ARP/ND requests to the target upon receiving a packet with destination IP not in its IP to MAC cache. When ARP/ND replies are received, the router will send the data packet to the target. This practice minimizes flooding when targets don't exist in the subnet.

When the target doesn't exist in the subnet, routers generally re-send an ARP/ND request a few more times before dropping the packets. So, the holding time by routers to wait for ARP/ND response can be longer than the time taken by the Pull Model to get IP to MAC mapping from a directory if target doesn't exist in the subnet.

For RBridges with mapping entries being pushed down from directory server, they can be configured to use Pull model for targets which don't exist in the mapping data pushed down.

A separate document will specify the detailed messages and mechanism for edge RBridges to pull information from directory server(s).

6. Recommendation

TRILL should provide a directory assisted approach. This document describes a basic framework of using a directory assisted approach for RBridge edge nodes. More detailed mechanisms will be described in a separate document or documents.

7. Security Considerations

Accurate mapping of IP addresses into MAC addresses and of MAC addresses to the RBridge from which they are reachable is important to the correct delivery of information. The security of specific directory assisted mechanisms will be discussed in the document or documents specifying those mechanisms.

For general TRILL security considerations, see [RFC6325].

8. IANA Considerations

This document requires no IANA actions. RFC Editor: please delete this section before publication.

9. Acknowledgements

Thanks for comments and review from the following:

David Black, Erik Nordmark

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

10. References

10.1 Normative References

As an Informational document, this draft has no Normative References.

10.2 Informative References

- [802.1Q] - IEEE Std 802.1Q-2011, "IEEE Standard for Local and metropolitan area networks - Virtual Bridged Local Area Networks", May 2011.
- [802.1Qbg] - IEEE Std 802.1Qbg-2012, "'Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks --- Edge Virtual Bridging'", July 2012.
- [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.
- [RFC826] - Plummer, D., "An Ethernet Address Resolution Protocol", RFC 826, November 1982.
- [RFC4861] - Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.
- [RFC6325] - Perlman, R., Eastlake 3rd, D., Dutt, D., Gai, S., and A. Ghanwani, "Routing Bridges (RBridges): Base Protocol Specification", RFC 6325, July 2011.
- [RFC6326] - Eastlake, D., Banerjee, A., Dutt, D., Perlman, R., and A. Ghanwani, "Transparent Interconnection of Lots of Links (TRILL) Use of IS-IS", RFC 6326, July 2011.

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
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 USA
Email: igor@yahoo-inc.com

Copyright, Disclaimer, and Additional IPR Provisions

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License. The definitive version of an IETF Document is that published by, or under the auspices of, the IETF. Versions of IETF Documents that are published by third parties, including those that are translated into other languages, should not be considered to be definitive versions of IETF Documents. The definitive version of these Legal Provisions is that published by, or under the auspices of, the IETF. Versions of these Legal Provisions that are published by third parties, including those that are translated into other languages, should not be considered to be definitive versions of these Legal Provisions. For the avoidance of doubt, each Contributor to the IETF Standards Process licenses each Contribution that he or she makes as part of the IETF Standards Process to the IETF Trust pursuant to the provisions of RFC 5378. No language to the contrary, or terms, conditions or rights that differ from or are inconsistent with the rights and licenses granted under RFC 5378, shall have any effect and shall be null and void, whether published or posted by such Contributor, or included with or in such Contribution.

TRILL Working Group
INTERNET-DRAFT
Intended Status: Informational

Samer Salam
Tissa Senevirathne
Cisco

Sam Aldrin
Donald Eastlake
Huawei

Expires: August 23, 2013

February 19, 2013

TRILL OAM Framework
draft-ietf-trill-oam-framework-01

Abstract

This document specifies a reference framework for Operations, Administration and Maintenance (OAM) in TRILL networks. The focus of the document is on the fault and performance management aspects of TRILL OAM.

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at
<http://www.ietf.org/lid-abstracts.html>

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

Copyright and License Notice

Copyright (c) 2013 IETF Trust and the persons identified as the

document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	4
1.1 Terminology	5
1.2 Relationship to Other OAM Work	5
2. TRILL OAM Model	6
2.1 OAM Layering	6
2.1.1 Relationship to CFM	7
2.1.2 Relationship to BFD	8
2.1.3 Relationship to Link OAM	8
2.2 TRILL OAM in the RBridge Port Model	9
2.3 Network, Service and Flow OAM	10
2.4 Maintenance Domains	11
2.5 Maintenance Entity and Maintenance Entity Group	12
2.6 MEPs and MIPs	12
2.7 Maintenance Point Addressing	14
3. OAM Frame Format	15
3.1 Motivation	15
3.2 Determination of Flow Entropy	16
3.2.1 Address Learning and Flow Entropy	17
3.3 OAM Message Channel	17
3.4 Identification of OAM Messages	17
4. Fault Management	17
4.1 Proactive Fault Management Functions	17
4.1.1 Fault Detection (Continuity Check)	18
4.1.2 Defect Indication	18
4.1.2.1 Forward Defect Indication	18
4.1.2.2 Reverse Defect Indication (RDI)	19
4.2 On-Demand Fault Management Functions	19
4.2.1 Connectivity Verification	19
4.2.1.1 Unicast	19
4.2.1.2 Multicast	20
4.2.2 Fault Isolation	21
5. Performance Management	21

5.1 Packet Loss	21
5.2 Packet Delay	22
6. Security Considerations	23
7. IANA Considerations	23
8. Acknowledgements	23
9. References	23
9.1 Normative References	23
9.2 Informative References	24
Authors' Addresses	25

1. Introduction

This document specifies a reference framework for Operations, Administration and Maintenance (OAM, [RFC6291]) in TRILL (Transparent Interconnection of Lots of Links) networks.

TRILL [RFC6325] specifies a protocol for shortest-path frame routing in multi-hop networks with arbitrary topologies and link technologies, using the IS-IS routing protocol. TRILL capable devices are referred to as TRILL Switches or RBridges (Routing Bridges). RBridges provide an optimized and transparent Layer 2 delivery service for Ethernet unicast and multicast traffic. Some characteristics of a TRILL network that are different from Ethernet bridging are the following:

- TRILL networks support arbitrary link technology between TRILL switches. Hence, a TRILL switch port may not have a 48-bit MAC Address [802] but might, for example, have an IP address as an identifier [TRILL-IP] or no unique identifier (PPP [RFC6361]).
- TRILL networks do not enforce congruency of unicast and multicast paths between a given pair of RBridges.
- TRILL networks do not impose symmetry of the forward and reverse paths between a given pair of RBridges.
- TRILL supports multipathing of unicast as well as multicast traffic.

In this document, we refer to the term OAM as defined in [RFC6291]. The Operations aspect involves finding problems that prevent proper functioning of the network. It also includes monitoring of the network to identify potential problems before they occur. Administration involves keeping track of network resources. Maintenance activities are focused on facilitating repairs and upgrades as well as corrective and preventive measures. [ISO/IEC 7498-4] defines 5 functional areas in the OSI model for network management, commonly referred to as FCAPS:

- Fault Management
- Configuration Management
- Accounting Management
- Performance Management
- Security Management

The focus of this document is on the first and fourth functional aspects, namely Fault Management and Performance Management, in TRILL networks. These primarily map to the "Operations" and "Maintenance"

part of OAM.

The draft provides a generic framework for a comprehensive solution that meets the requirements outlined in [TRILL-OAM-REQ]. However, specific mechanisms to address these requirements are considered to be outside the scope of this document.

1.1 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].

In addition, the following acronyms are used:

- BFD - Bidirectional Forwarding Detection [RFC5880]
- CFM - Connectivity Fault Management [802.1Q]
- FGL - Fine Grained Label(ing) [TRILL-FGL]
- IEEE - Institute for Electrical and Electronic Engineers
- IP - Internet Protocol, includes both IPv4 and IPv6
- L2VPN - Layer 2 Virtual Private Network
- LAN - Local Area Network
- MEG - Maintenance Entity Group
- MEP - Maintenance End Point
- MIP - Maintenance Intermediate Point
- MP - Maintenance Point (MEP or MIP)
- OAM - Operations, Administration, and Maintenance [RFC6291]
- RBridge - Routing Bridge, a device implementing TRILL [RFC6325]
- TRILL - Transparent Interconnection of Lots of Links [RFC6325]
- TRILL Switch - an alternate name for an RBridge
- VLAN - Virtual LAN

1.2 Relationship to Other OAM Work

OAM is a technology area where a wealth of prior art exists. This document leverages concepts and draws upon elements defined and/or used in the following documents:

[TRILL-OAM-REQ] defines the requirements for TRILL OAM that serve as the basis for this framework.

[802.1Q] specifies the Connectivity Fault Management protocol, which defines the concepts of Maintenance Domains, Maintenance End Points, and Maintenance Intermediate Points.

[Y.1731] extends Connectivity Fault Management in the following areas: it defines fault notification and alarm suppression functions for Ethernet. It also specifies mechanisms for Ethernet performance management, including loss, delay, jitter, and throughput

measurement.

[RFC6136] specifies a reference model for OAM as it relates to L2VPN services, pseudowires and associated Public Switched Network tunnels. The document also specifies OAM requirements for L2VPN services.

[RFC6371] describes a framework to support a comprehensive set of OAM procedures that fulfill the MPLS-TP OAM requirements for fault, performance, and protection-switching management and that do not rely on the presence of a control plane.

[TRILL-BFD] defines a TRILL encapsulation for BFD that enables the use of the latter for network fast convergence.

2. TRILL OAM Model

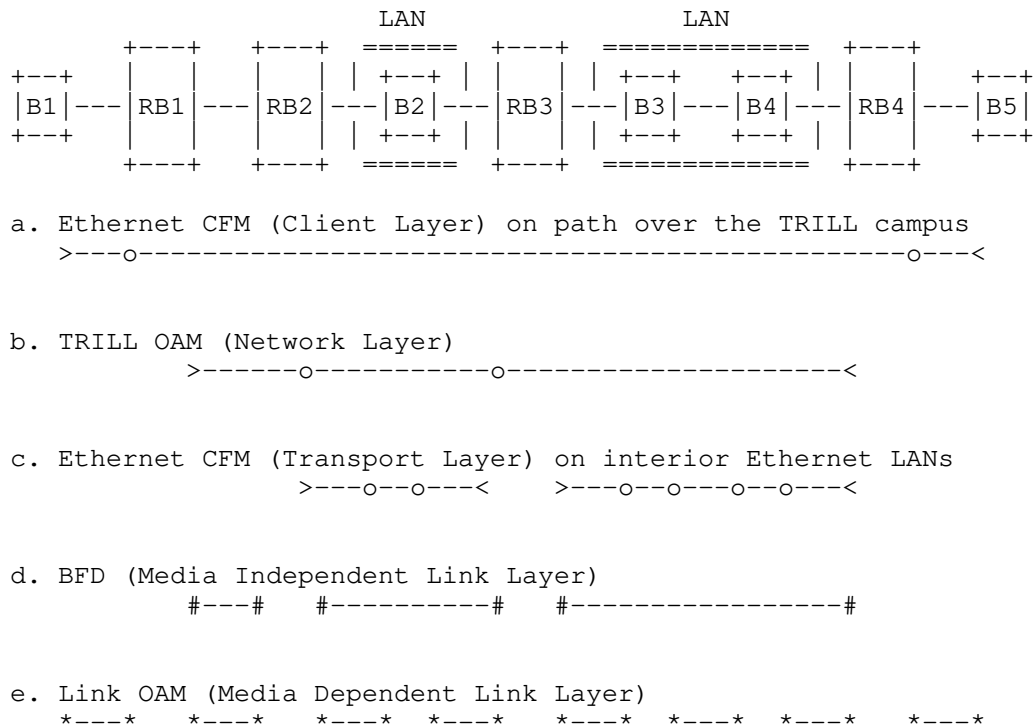
2.1 OAM Layering

In the TRILL architecture, the TRILL layer is independent of the underlying Link Layer technology. Therefore, it is possible to run TRILL over any transport layer capable of carrying TRILL frames such as Ethernet [RFC6325], PPP [RFC6361], or IP [TRILL-IP]. Furthermore, TRILL provides a virtual Ethernet connectivity service that is transparent to higher layer entities (e.g. Layer 3 and above). This strict layering is observed by TRILL OAM.

Of particular interest is the layering of TRILL OAM with respect to:

- BFD, which is typically used for fast convergence
- Ethernet CFM [802.1Q] on paths from an external device, over a TRILL campus, to another external device, especially since TRILL switches are likely to be deployed where existing 802.1 bridges can be such external devices.
- Link OAM, on links interior to a TRILL campus, which is link technology specific.

Consider the example network depicted in Figure 1 below, where a TRILL network is interconnected via Ethernet links:



Legend: > MEP o MIP # BFD Endpoint * Link OAM Endpoint

Figure 1: OAM Layering in TRILL

Where B_n and R_{Bn} (n= 1,2,3, ...) denote IEEE 802.1Q bridges and TRILL Rbridges, respectively.

2.1.1 Relationship to CFM

In the context of a TRILL network, CFM can be used as either a client layer OAM or a transport layer OAM mechanism.

When acting as a client layer OAM (see Figure 1a), CFM provides fault management capabilities for the user, on an end-to-end basis over the TRILL network. Edge ports of the TRILL network may be visible to CFM operations through the optional presence of a CFM Maintenance Intermediate Point (MIP) in the TRILL switches edge Ethernet ports.

When acting as a transport layer OAM (see Figure 1c), CFM provides fault management functions for the IEEE 802.1Q bridged LANs that may interconnect Rbridges. Such bridged LANs can be used as TRILL level

links between RBridges. RBridges directly connected to the intervening 802.1Q bridges may host CFM Down Maintenance End Points (MEPs).

2.1.2 Relationship to BFD

One-hop BFD (see Figure 1d) runs between adjacent RBridges and provides fast link as well as node failure detection capability [TRILL-BFD]. Note that BFD sits a layer above Link OAM, which is media specific. BFD provides fast convergence characteristics to TRILL networks. It is worth noting that the requirements for BFD are different from those of the TRILL OAM mechanisms that are the prime focus of this document. Furthermore, BFD does not use the frame format described in section 3.1.

TRILL BFD differs from TRILL OAM in two significant ways:

1. A TRILL BFD transmitter is bound to a specific TRILL output port as explained below.
2. TRILL BFD messages can be transmitted by the originator out a port to a neighbor RBridge when the adjacency is in the Detect or Two-Way states as well as when the adjacency is in the Up state [RFC6327].

In contrast, TRILL OAM messages are initially transmitted by appearing to have been received on a TRILL input port (refer to Section 2.2 for details). The output ports on which TRILL OAM message are sent are determined by the TRILL routing function, which will only send on links that are in the Up state and have been incorporated into the local view of the campus topology.

For example, assume there are five parallel equal cost links between RB1 and RB2 that have not been aggregated. (Links that are aggregated with [802.1AX] appear to TRILL to be a single link accessible through a single TRILL port.) However, RB1 is only capable of doing up to 4-way ECMP. TRILL OAM messages, as dispatched by the TRILL Routing function, will use 4 of the 5 links. But it is desirable to be able to monitor the fifth link to be sure it is available for failover. TRILL BFD messages sent by RB1 will use the output port to which their session is bound. RB1 can easily monitor all 5 links to RB2 by using a TRILL BFD session bound to each of the 5 output ports.

2.1.3 Relationship to Link OAM

Link OAM (see Figure 1e) depends on the nature of the technology used in the links interconnecting RBridges. For e.g., for Ethernet links, [802.3] Clause 57 OAM may be used.

2.2 TRILL OAM in the RBridge Port Model

TRILL OAM processing can be modeled as a layer situated between the port's TRILL encapsulation/de-capsulation function and the RBridge Forwarding Engine function, on any RBridge port. TRILL OAM requires services of the RBridge forwarding engine and utilizes information from the IS-IS control plane. Figure 2 below depicts TRILL OAM processing in the context of the RBridge port model defined in [RFC6325]. In this figure, double lines represent flow of both frames and information.

While this figure shows a conceptual model, it is to be understood that implementations need not mirror this exact model as long as the intended OAM requirements and functionality are preserved.

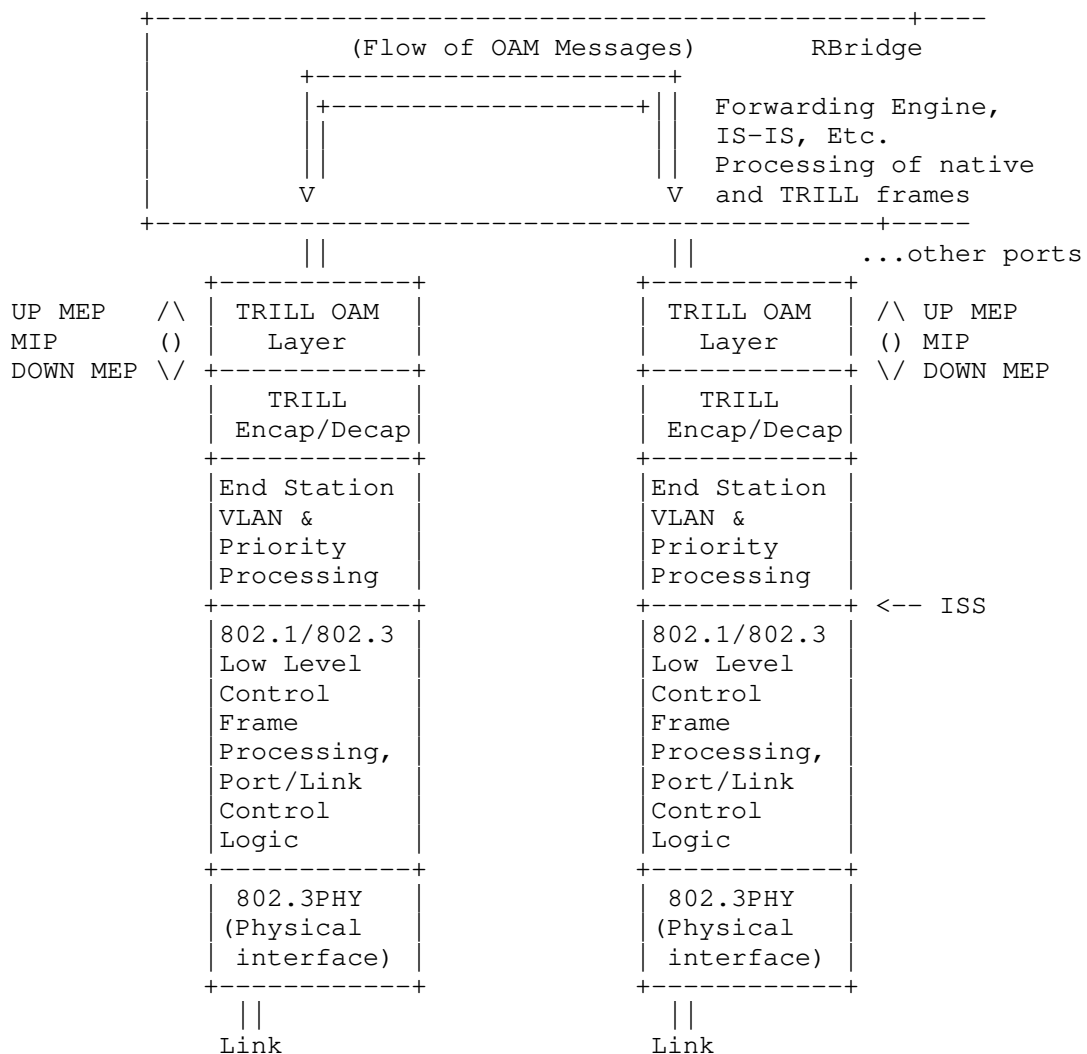


Figure 2: TRILL OAM in RBridge Port Model

Note that the terms "MEP" and "MIP" in the above figure are explained in detail in section 2.6 below.

2.3 Network, Service and Flow OAM

OAM functions in a TRILL network can be conducted at different levels of granularity. This gives rise to 'Network', 'Service' and 'Flow'

OAM, listed in order of increasing granularity.

Network OAM mechanisms provide fault and performance management functions in the context of a representative 'test' VLAN or fine-grained label [TRILL-FGL]. The test VLAN can be thought of as a management or diagnostics VLAN which extends to all RBridges in a TRILL network. In order to account for multipathing, Network OAM functions also make use of test flows (both unicast and multicast) to provide coverage of the various paths in the network.

Service OAM mechanisms provide fault and performance management functions in the context of the actual VLAN or fine-grained label set for which end station service is enabled. Test flows are used here, as well, to provide coverage in the case of multipathing.

Flow OAM mechanisms provide the most granular fault and performance management capabilities, where OAM functions are performed in the context of end station service VLANs or fine grained labels and user flows. While Flow OAM provides the most granular control, it clearly poses scalability challenges if attempted on large numbers of flows.

2.4 Maintenance Domains

The concept of Maintenance Domains, or OAM Domains, is well known in the industry. IEEE [802.1Q], [RFC6136], [RFC5654], etc... all define the notion of a Maintenance Domain as a collection of devices (e.g. network elements) that are grouped for administrative and/or management purposes. Maintenance domains usually delineate trust relationships, varying addressing schemes, network infrastructure capabilities, etc...

When mapped to TRILL, a Maintenance Domain is defined as a collection of RBridges in a network for which faults in connectivity or performance are to be managed by a single operator. All RBridges in a given Maintenance Domain are, by definition, managed by a single entity (e.g. an enterprise or a data center operator, etc...). [RFC6325] defines the operation of TRILL in a single IS-IS area, with the assumption that a single operator manages the network. In this context, a single (default) Maintenance Domain is sufficient for TRILL OAM.

However, when considering scenarios where different TRILL networks need to be interconnected, for e.g. as discussed in [TRILLML], then the introduction of multiple Maintenance Domains and Maintenance Domain hierarchies becomes useful to map and contain administrative boundaries. When considering multi-domain scenarios, the following rules must be followed: TRILL OAM domains MUST NOT overlap, but MUST

either be disjoint or nest to form a hierarchy (i.e. a higher Maintenance Domain MAY completely engulf a lower Domain). A Maintenance Domain is typically identified by a Domain Name and a Maintenance Level (a numeric identifier). The larger the Domain, the higher the Level number.

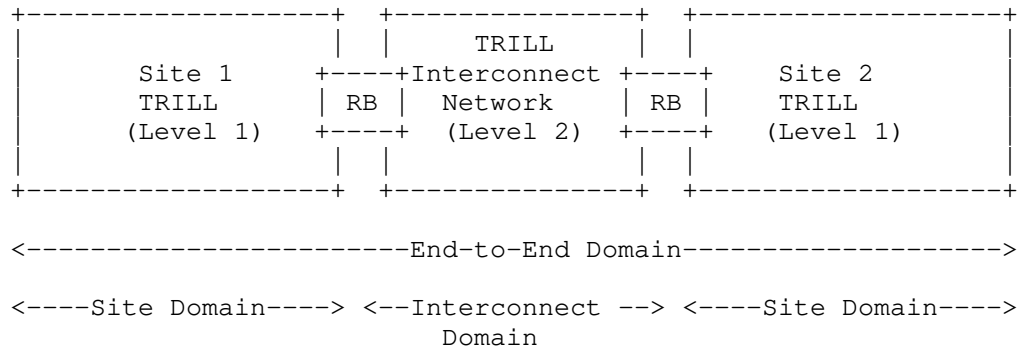


Figure 3: TRILL OAM Maintenance Domains

2.5 Maintenance Entity and Maintenance Entity Group

TRILL OAM functions are performed in the context of logical endpoint pairs referred to as Maintenance Entities (ME). A Maintenance Entity defines a relationship between two points in a TRILL network where OAM functions (e.g. monitoring operations) are applied. The two points that define a Maintenance Entity are known as Maintenance End Points (MEPs) – see section 2.6 below. The set of Maintenance Entities that belong to the same Maintenance Domain are referred to as a Maintenance Entity Group (MEG). On the network path in between MEPs, there can be zero or more intermediate points, called Maintenance Intermediate Points (MIPs). MEPs and MIPs are associated with the MEG and can be part of more than one ME in a given MEG.

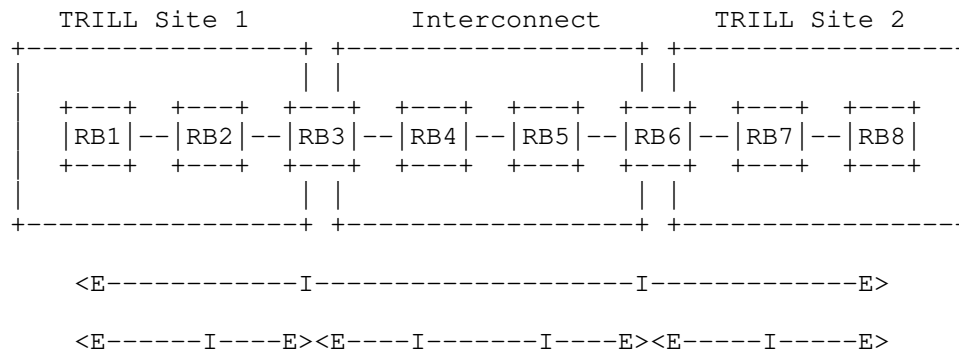
2.6 MEPs and MIPs

OAM capabilities on RBridges can be defined in terms of logical groupings of functions that can be categorized into two functional objects: Maintenance End Points (MEPs) and Maintenance Intermediate Points (MIPs). The two are collectively referred to as Maintenance Points (MPs).

MEPs are the active components of TRILL OAM: MEPs source TRILL OAM messages proactively or on-demand based on operator invocation. Furthermore, MEPs ensure that TRILL OAM messages do not leak outside a given Maintenance Domain, e.g. out of the TRILL network and into end stations. MIPs, on the other hand, are internal to a Maintenance

Domain. They are the more passive components of TRILL OAM, primarily responsible for forwarding TRILL OAM messages and selectively responding to a subset of these messages.

The following figure shows the MEP and MIP placement for the Maintenance Domains depicted in Figure 3 above.



Legend E: MEP I: MIP

Figure 4: MEPs and MIPs

It is worth noting that a single RBridge may host multiple MEPs of different technologies, e.g. TRILL OAM MEP(s) and [802.1Q] MEP(s). This does not mean that the protocol operation is necessarily consolidated into a single functional entity on those ports. The protocol functions for each MEP remain independent and reside in different shims in the RBridge Port model of Figure 2: the TRILL OAM MEP resides in the "TRILL OAM Processing" block whereas a CFM MEP resides in the "802.1Q Port VLAN Processing" block.

In the model of Section 2.2, a single MEP and/or MIP per MEG can be instantiated per RBridge port. A MEP is further qualified with an administratively set direction (UP or DOWN), as follows:

- An UP MEP sends and receives OAM messages through the RBridge Forwarding Engine. This means that an UP MEP effectively communicates with MEPs on other RBridges through TRILL interfaces other than the one that the MEP is configured on.
- A DOWN MEP sends and receives OAM messages through the link connected to the interface on which the MEP is configured.

In order to support TRILL OAM functions on sections, as specified in [TRILL-OAM-REQ], while maintaining the simplicity of a single TRILL OAM Maintenance Domain, the TRILL OAM Layer may be implemented on a virtual port with no physical layer (Null PHY). In this case, the Down MEP function is not supported, since the virtual port does not attach to a link; as such, a Down MEP would not be capable of sending or receiving OAM messages.

A TRILL OAM solution that conforms to this framework:

- MUST support the MIP function on TRILL physical ports (to support fault isolation)
- MUST support the UP MEP function on a TRILL virtual port (to support OAM functions on Sections)
- MAY support the UP MEP function on TRILL physical ports
- MAY support the DOWN MEP function on TRILL physical ports

2.7 Maintenance Point Addressing

TRILL OAM functions must provide the capability to address a specific Maintenance Point or a set of one or more Maintenance Points in a MEG. To that end, RBridges need to recognize two sets of addresses:

- Individual MP addresses
- Group MP Addresses

TRILL OAM will support the Shared MP address model, where all MPs on an RBridge share the same Individual MP address. In other words, TRILL OAM messages can be addressed to a specific RBridge but not to a specific port on an RBridge.

One cannot discern, from observing the external behavior of an RBridge, whether TRILL OAM messages are actually delivered to a certain MP or another entity within the RBridge. The Shared MP address model takes advantage of this fact by allowing MPs in different RBridge ports to share the same Individual MP address. The MPs may still be implemented as residing on different RBridge ports and for the most part, they have distinct identities.

The Group MP addresses enable the OAM mechanism to reach all the MPs in a given MEG. Certain OAM functions, e.g. pruned tree verification, require addressing a subset of the MPs in a MEG. Group MP addresses are not defined for such subsets. Rather, the OAM function in question must use the Group MP addresses combined with an indication of the scope of the MP subset encoded in the OAM Message Channel. This prevents the unwieldy response to Group MP addresses.

3. OAM Frame Format

3.1 Motivation

In order for TRILL OAM messages to accurately test the data-path, these messages must be transparent to transit RBridges. That is, a TRILL OAM message must be indistinguishable from a TRILL data frame through normal transit RBridge processing. Only the target RBridge, which needs to process the message, should identify and trap the packet as a control message through normal processing. Additionally methods must be provided to prevent OAM packets from being transmitted out as native frames.

The TRILL OAM frame format proposed below provides the necessary flexibility to exercise the data path as closely as possible to actual data packets.

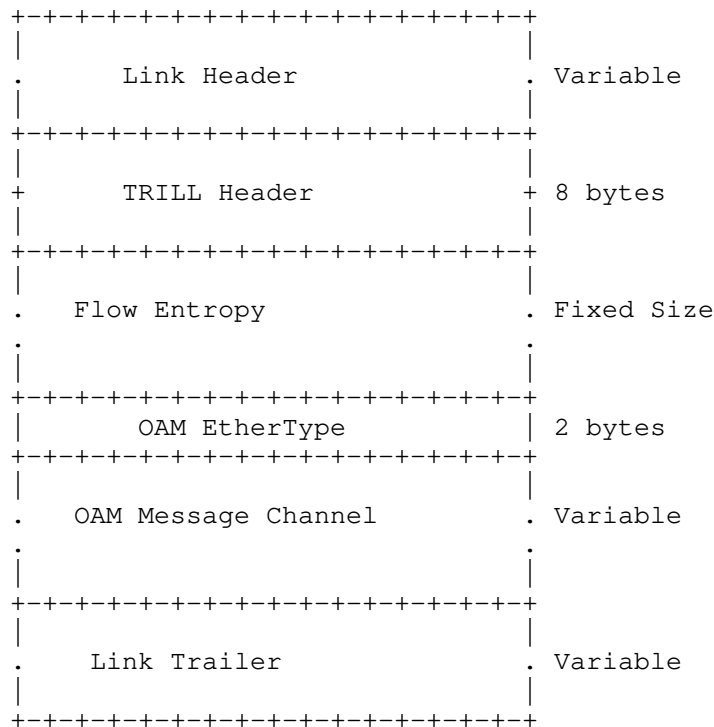


Figure 5: OAM Frame Format

The TRILL Header is as specified in [RFC6325] and the Link Header and Trailer are as specified for the link technology. (Link types standardized so far are [RFC6325] for Ethernet and [RFC6361] for PPP). These fields need to be as similar as practical to the Link Header/Trailer and TRILL Header of the normal TRILL data frame corresponding to the traffic that OAM is testing.

The OAM EtherType demarcates the boundary between the Flow Entropy and the OAM Message Channel. The OAM EtherType is expected at a deterministic offset from the TRILL Header, thereby allowing applications to clearly identify the beginning of the OAM Message Channel. Additionally, it facilitates the use of the same OAM frame structure by different Ethernet technologies.

The Link Trailer is usually a checksum, such as the Ethernet Frame Check Sequence, which is examined at a low level very early in the frame input process and automatically generated as part of the low level frame output process. If the checksum fails, the frame is normally discarded with no higher level processing.

3.2 Determination of Flow Entropy

The Flow Entropy is a fixed length field that is populated with either real packet data or synthetic data that mimics the intended flow.

For a Layer 2 flow (i.e. non-IP) the Flow Entropy must specify the Ethernet header, including the MAC destination and source addresses as well as a VLAN tag or fine grain label.

For a Layer 3 flow, the Flow Entropy must specify the Ethernet header, the IP header and UDP or TCP header fields.

Not all fields in the Flow Entropy field need to be identical to the data flow that the OAM message is mimicking. The only requirement is for the selected flow entropy to follow the same path as the data flow that it is mimicking. In other words, the selected flow entropy must result in the same ECMP selection or multicast pruning behavior or other applicable forwarding paradigm.

When performing diagnostics on user flows, the OAM mechanisms must allow the network operator to configure the flow entropy parameters (e.g. Layer 2 and/or 3) on the RBridge from which the diagnostic operations are to be triggered.

When running OAM functions over Test Flows, the TRILL OAM should provide a mechanism for discovering the flow entropy parameters by querying the RBridges dynamically.

3.2.1 Address Learning and Flow Entropy

Edge TRILL switches, like traditional 802.1 bridges, are required to learn MAC address associations. Learning is accomplished either by snooping data packets or through other methods. The flow entropy field of TRILL OAM messages mimics real packets and may impact the address learning process of the TRILL data plane. TRILL OAM is required to provide methods to prevent any learning of addresses from the flow entropy field of OAM messages that would interfere with normal TRILL operation. This can be done, for e.g., by suppressing/preventing MAC address learning from OAM messages.

3.3 OAM Message Channel

The OAM Message Channel provides methods to communicate OAM specific details between RBridges. [802.1Q] CFM and [RFC4379] have implemented OAM message channels. It is desirable to select an appropriate technology and re-use it, instead of redesigning yet another OAM channel. TRILL is a transport layer that carries Ethernet frames, so the TRILL OAM model specified earlier is based on the [802.1Q] CFM model. The use of [802.1Q] CFM encoding format for the OAM Message channel is one possible choice. [TRILL-OAM] presents a proposal on the use of [802.1Q] CFM payload as the OAM message channel.

3.4 Identification of OAM Messages

RBridges must be able to identify OAM messages that are destined to them, either individually or as a group, so as to properly process those messages.

It may be possible to use a combination of one of the unused fields or bits in the TRILL Header and the OAM EtherType to identify TRILL OAM messages.

[RFC6325] does not specify any method of identifying OAM messages. Hence, for backwards compatibility reasons, TRILL OAM solutions must provide methods to identify OAM messages through the use of well-known patterns in the Flow Entropy field; for e.g., by using a reserved MAC address as the inner MAC SA.

4. Fault Management

Section 4.1 below discusses proactive fault management and Section 4.2 discusses on-demand fault management.

4.1 Proactive Fault Management Functions

Proactive fault management functions are configured by the network

operator to run periodically without a time bound, or are configured to trigger certain actions upon the occurrence of specific events.

4.1.1 Fault Detection (Continuity Check)

Proactive fault detection is performed by periodically monitoring the reachability between service endpoints, i.e. MEPs in a given MEG, through the exchange of Continuity Check messages. The reachability between any two arbitrary MEP may be monitored for a specified path, all paths or any representative path. The fact that TRILL networks do not enforce congruency between unicast and multicast paths means that the proactive fault detection mechanism must provide procedures to monitor the unicast paths independently of the multicast paths. Furthermore, where the network has ECMP, the proactive fault detection mechanism must be capable of exercising the equal-cost paths individually.

The set of MEPs exchanging Continuity Check messages in a given domain and for a specific monitored entity (flow, network or service) must use the same transmission period. As long as the fault detection mechanism involves MEPs transmitting periodic heartbeat messages independently, then this OAM procedure is not affected by the lack of forward/reverse path symmetry in TRILL.

The proactive fault detection function must detect the following types of defects:

- Loss of continuity (LoC) to one or more remote MEPs
- Unexpected connectivity between isolated VLANs (mismatch)
- Unexpected connectivity to one or more remote MEPs
- Period mis-configuration

4.1.2 Defect Indication

TRILL OAM MUST support event-driven defect indication upon the detection of a connectivity defect. Defect indications can be categorized into two types:

4.1.2.1 Forward Defect Indication

This is used to signal a failure that is detected by a lower layer OAM mechanism. Forward Defect indication is transmitted away from the direction of the failure. For e.g., consider a simple network comprising of four RBridges connected in tandem: RB1, RB2, RB3 and RB4. Both RB1 and RB4 are hosting TRILL OAM MEPs, whereas RB2 and RB3 have MIPs. If the link between RB2 and RB3 fails, then RB2 can send a forward defect indication towards RB1 while RB3 sends a forward defect indication towards RB4.

Forward defect indication may be used for alarm suppression and/or for purpose of inter-working with other layer OAM protocols. Alarm suppression is useful when a transport/network level fault translates to multiple service or flow level faults. In such a scenario, it is enough to alert a network management station (NMS) of the single transport/network level fault in lieu of flooding that NMS with a multitude of Service or Flow granularity alarms.

4.1.2.2 Reverse Defect Indication (RDI)

RDI is used to signal that the advertising MEP has detected a loss of continuity (LoC) defect. RDI is transmitted in the direction of the failure. For e.g., consider the same tandem network of the previous section. If RB1 detects that it has lost connectivity to RB4 because it is no longer receiving Continuity Check messages from the MEP on RB4, then RB1 can transmit an RDI towards RB4 to inform the latter of the failure. If the failure is unidirectional (i.e. it is affecting the direction from RB4 to RB1), then the RDI enables RB4 to become aware of the unidirectional connectivity anomaly.

RDI allows single-sided management, where the network operator can examine the state of a single MEP and deduce the overall health of a monitored entity (network, flow or service).

4.2 On-Demand Fault Management Functions

On-demand fault management functions are initiated manually by the network operator and continue for a time bound period. These functions enable the operator to run diagnostics to investigate a defect condition.

4.2.1 Connectivity Verification

As specified in [TRILL-OAM-REQ], TRILL OAM must support on-demand connectivity verification for unicast and multicast. The connectivity verification mechanism must provide a means for specifying and carrying in the messages:

- variable length payload/padding to test MTU related connectivity problems.
- test traffic patterns as defined in [RFC2544].

4.2.1.1 Unicast

Unicast connectivity verification operation must be initiated from a MEP and may target either a MIP or another MEP. For unicast, connectivity verification can be performed at either Network or Flow

granularity.

Connectivity verification at the Network granularity tests connectivity between a MEP on a source RBridge and a MIP or MEP on a target RBridge over a representative test VLAN and for a test flow. The operator must supply the source and target RBridges for the operation, and the test VLAN/flow information uses pre-set values or defaults.

Connectivity verification at the Flow granularity tests connectivity between a MEP on a source RBridge and a MIP or MEP on a target RBridge over an operator specified VLAN or fine grain label with operator specified flow parameters.

The above functions must be supported on sections, as defined in [TRILL-OAM-REQ]. When connectivity verification is triggered over a section, and the initiating MEP does not coincide with the edge (ingress) RBridge, the MEP must use the edge RBridge nickname instead of the local RBridge nickname on the associated connectivity verification messages. The operator must supply the edge RBridge nickname as part of the operation parameters.

4.2.1.2 Multicast

For multicast, the connectivity verification function tests all branches and leaf nodes of a multidestination distribution tree for reachability. This function should include mechanisms to prevent reply storms from overwhelming the initiating RBridge. This may be done, for e.g., by staggering the replies. To further prevent reply storms, connectivity verification operation is initiated from a MEP and must target MEPs only. MIPs are transparent to multicast connectivity verification.

Per [TRILL-OAM-REQ], multicast connectivity verification must provide the following granularity of operation:

A. Un-pruned Tree

- Connectivity verification for un-pruned multidestination distribution tree. The operator in this case supplies the tree identifier (root RBridge nickname) and campus wide diagnostic VLAN.

B. Pruned Tree

- Connectivity verification for a VLAN or fine-grain label in a given multidestination distribution tree. The operator in this case supplies the tree identifier and VLAN or fine grain label.

- Connectivity verification for an IP multicast group in a given multideestination distribution tree. The operator in this case supplies: the tree identifier, VLAN or fine grain label and IP (S,G) or (*,G).

4.2.2 Fault Isolation

TRILL OAM must support an on-demand connectivity fault localization function. This is the capability to trace the path of a Flow on a hop-by-hop (i.e. RBridge by RBridge) basis to isolate failures. This involves the capability to narrow down the locality of a fault to a particular port, link or node. The characteristic of forward/reverse path asymmetry, in TRILL, renders fault isolation into a direction-sensitive operation. That is, given two RBridges A and B, localization of connectivity faults between them requires running fault isolation procedures from RBridge A to RBridge B as well as from RBridge B to RBridge A. Generally speaking, single-sided fault isolation is not possible in TRILL OAM.

5. Performance Management

Performance Management functions can be performed both proactively and on-demand. Proactive management involves a scheduling function, where the performance management probes can be triggered on a recurring basis. Since the basic performance management functions involved are the same, we make no distinction between proactive and on-demand functions in this section.

5.1 Packet Loss

Given that TRILL provides inherent support for multipoint-to-multipoint connectivity, then packet loss cannot be accurately measured by means of counting user data packets. This is because user packets can be delivered to more RBridges or more ports than are necessary (e.g. due to broadcast, un-pruned multicast or unknown unicast flooding). As such, a statistical means of approximating packet loss rate is required. This can be achieved by sending "synthetic" (i.e. TRILL OAM) packets that are counted only by those ports (MEPs) that are required to receive them. This provides a statistical approximation of the number of data frames lost, even with multipoint-to-multipoint connectivity.

Packet loss probes must be initiated from a MEP and must target a MEP. This function must be supported on sections, as defined in [TRILL-OAM-REQ]. When packet loss is measured over a section, and the initiating MEP does not coincide with the edge (ingress) RBridge, the MEP must use the edge RBridge nickname instead of the local RBridge nickname on the associated loss measurement messages. The user must

supply the edge RBridge nickname as part of the operation parameters.

5.2 Packet Delay

Packet delay is measured by inserting time-stamps in TRILL OAM packets. In order to ensure high accuracy of measurement, TRILL OAM must specify the time-stamp location at fixed offsets within the OAM packet in order to facilitate hardware-based time-stamping. Hardware implementations must implement the time-stamping function as close to the wire as practical in order to maintain high accuracy.

6. Security Considerations

TRILL OAM must provide mechanisms for:

- Preventing denial of service attacks caused by exploitation of the OAM message channel.
- Optionally authenticate communicating endpoints (MEPs and MIPs)
- Preventing TRILL OAM packets from leaking outside of the TRILL network or outside their corresponding Maintenance Domain. This can be done by having MEPs implement a filtering function based on the Maintenance Level associated with received OAM packets.

For general TRILL Security Considerations, see [RFC6325].

7. IANA Considerations

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

8. Acknowledgements

We invite feedback and contributors.

9. References

9.1 Normative References

- [TRILL-OAM-REQ] Senevirathne, "Requirements for Operations, Administration and Maintenance (OAM) in TRILL", draft-tissa-trill-oam-req, work in progress.
- [RFC6325] Perlman, et al., "Routing Bridges (RBriges): Base Protocol Specification", RFC 6325, July 2011.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC6136] Sajassi, A., Ed., and D. Mohan, Ed., "Layer 2 Virtual Private Network (L2VPN) Operations, Administration, and Maintenance (OAM) Requirements and Framework", RFC 6136, March 2011.
- [RFC2544] Bradner, S. and J. McQuaid, "Benchmarking Methodology for Network Interconnect Devices", RFC 2544, March 1999.

- [RFC6291] Andersson et al., BCP 161 "Guidelines for the Use of the "OAM" Acronym in the IETF", June 2011.
- [RFC6327] Eastlake 3rd, D., Perlman, R., Ghanwani, A., Dutt, D., and V. Manral, "Routing Bridges (RBridges): Adjacency", RFC 6327, July 2011.
- [TRILL-FGL] D. Eastlake et al., "TRILL Fine-Grained Labeling", draft-ietf-trill-fine-labeling, work in progress.
- [802.1Q] "IEEE Standard for Local and metropolitan area networks - Media Access Control (MAC) Bridges and Virtual Bridge Local Area Networks", IEEE Std 802.1Q-2011, 31 August 2011.
- [RFC6371] Busi & Allan, "Operations, Administration, and Maintenance Framework for MPLS-Based Transport Networks", RFC 6371, September 2011.
- [802] "IEEE Standard for Local and Metropolitan Area Networks - Overview and Architecture", IEEE Std 802-2001, 8 March 2002.

9.2 Informative References

- [Y.1731] "ITU-T Recommendation Y.1731 (02/08) - OAM functions and mechanisms for Ethernet based networks", February 2008.
- [ISO/IEC 7498-4] "Information processing systems -- Open Systems Interconnection -- Basic Reference Model -- Part 4: Management framework", ISO/IEC, 1989.
- [TRILL-BFD] V. Manral, et al., "TRILL (Transparent Interconnection of Lots of Links): Bidirectional Forwarding Detection (BFD) Support", draft-ietf-trill-rbridge-bfd, work in progress, June 2012.
- [TRILL-OAM] T. Senevirathne, et al., "Use of 802.1ag for TRILL OAM Messages", draft-tissa-trill-8021ag, work in progress, June 2012.
- [TRILL-IP] M. Wasserman, et al., "Transparent Interconnection of Lots of Links (TRILL) over IP", draft-mrw-trill-over-ip, work in progress, September 2012.

Authors' Addresses

Samer Salam
Cisco
595 Burrard Street, Suite 2123
Vancouver, BC V7X 1J1, Canada
Email: ssalam@cisco.com

Tissa Senevirathne
Cisco
375 East Tasman Drive
San Jose, CA 95134, USA
Email: tsenevir@cisco.com

Sam Aldrin
Huawei Technologies
2330 Central Expressway
Santa Clara, CA 95050, USA
Email: sam.aldrin@gmail.com

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

TRILL Working Group
Internet Draft
Intended status: Standards Track
Expires: August 2013

T. Mizrahi
Marvell
T. Senevirathne
S. Salam
Cisco
D. Eastlake 3rd
Huawei
February 18, 2013

Loss and Delay Measurement in
Transparent Interconnection of Lots of Links (TRILL)
draft-mizrahi-trill-loss-delay-00.txt

Abstract

Performance Monitoring (PM) is a key aspect of Operations, Administration and Maintenance (OAM). It allows network operators to verify the Service Level Agreement (SLA) provided to customers, and to detect network anomalies. This document specifies mechanisms for Loss Measurement (LM) and Delay Measurement (DM) in TRILL networks.

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

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

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

This Internet-Draft will expire on August 18, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Conventions Used in this Document	4
2.1. Keywords	4
2.2. Definitions	4
2.3. Abbreviations	5
3. Loss and Delay Measurement in the TRILL Architecture	5
3.1. Performance Monitoring Granularity	6
3.2. One-Way vs. Two-Way Performance Monitoring	6
3.2.1. One-Way Performance Monitoring	7
3.2.2. Two-Way Performance Monitoring	7
3.3. Point-to-point PM vs. Point-to-multipoint PM	8
4. Loss Measurement	8
4.1. One-Way Loss Measurement (OWLM)	8
4.1.1. 1SLM Message Transmission	9
4.1.2. 1SLM Message Reception	9
4.2. Two-Way Loss Measurement (TWLM)	10
4.2.1. SLM Message Transmission	11
4.2.2. SLM Message Reception	12
4.2.3. SLR Message Reception	13
5. Delay Measurement	14
5.1. One-Way Delay Measurement (OWDM)	14
5.1.1. 1DM Message Transmission	15
5.1.2. 1DM Message Reception	15
5.2. Two-Way Delay Measurement (TWDM)	15
5.2.1. DMM Message Transmission	16
5.2.2. DMM Message Reception	17
5.2.3. DMR Message Reception	17
6. Packet Formats	18
6.1. TRILL OAM Encapsulation	18
6.2. Loss Measurement Packet Formats	20

6.2.1. Counter Format	20
6.2.2. 1SLM Packet Format	21
6.2.3. SLM Packet Format	22
6.2.4. SLR Packet Format	23
6.3. Delay Measurement Packet Formats	24
6.3.1. Timestamp Format	24
6.3.2. 1DM Packet Format	24
6.3.3. DMM Packet Format	25
6.3.4. DMR Packet Format	26
6.4. Reflector Entropy TLV	27
7. Security Considerations	27
8. IANA Considerations	27
8.1. OpCode Values	27
8.2. TLV Type	28
9. Acknowledgments	28
10. References	28
10.1. Normative References	28
10.2. Informative References	28

1. Introduction

TRILL [RFC6179] is a protocol for transparent least cost routing, where Rbridges forward traffic to their destination based on a least cost route, using a TRILL encapsulation header with a hop count.

Operations, Administration and Maintenance (OAM) [OAM] is a set of tools for detecting, isolating and reporting connection failures and performance degradation. Performance Monitoring (PM) is a key aspect of OAM. PM allows network operators to detect and debug network anomalies and incorrect behavior. PM consists of two main building blocks - Loss Measurement (LM) and Delay Measurement (DM). PM may also include other derived metrics such as Packet Delivery Rate (PDR), and delay variation.

The requirements of OAM in TRILL networks are defined in [OAM-REQ], and the TRILL OAM framework is described in [OAM-FRAMEWK]. These two documents also highlight the main requirements in terms of performance monitoring.

This document defines protocols for loss measurement and for delay measurement in TRILL networks. These protocols are somewhat based on the ones defined in [Y.1731].

- o Loss Measurement (LM): the LM protocol measures the packet loss between two RBridges. The measurement is performed by sending a set of synthetic packets, and counting the number of packets transmitted and received during the test. The loss rate is calculated by comparing the counters of transmitted and received packets.

This document does not define an LM protocol that computes the packet loss of data-plane traffic. For further details see [OAM-FRAMEWK].

- o Delay Measurement (DM): the DM protocol measures the packet delay and packet delay variation between two RBridges. The measurement is performed using timestamped OAM messages.

2. Conventions Used in this Document

2.1. Keywords

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 [KEYWORDS].

2.2. Definitions

- o One-way packet delay - (as defined in [OAM-REQ]) the time elapsed from the start of transmission of the first bit of a packet by an RBridge until the reception of the last bit of the packet by the remote RBridge.
- o Two-way packet delay - (as defined in [OAM-REQ]) the time elapsed from the start of transmission of the first bit of a packet from the local RBridge, receipt of the packet at the remote RBridge, the remote RBridge sending a response packet back to the local RBridge and the local RBridge receiving the last bit of that response packet.
- o Packet loss - the number of packets lost in a specific LM test, and a specific observation period.
- o Far-end packet loss - the number of packets lost on the path from the local RBridge to the remote RBridge in a specific LM test, and a specific observation period.
- o Near-end packet loss - the number of packets lost on the path from the remote RBridge to the local RBridge in a specific LM test, and a specific observation period.

2.3. Abbreviations

1DM	One-way Delay Measurement message
1LM	One-way Loss Measurement message
DM	Delay Measurement
DMM	Delay Measurement Message
DMR	Delay Measurement Reply
MD	Maintenance Domain
MD-L	Maintenance Domain Level
MEP	Maintenance End Point
MIP	Maintenance Intermediate Point
MP	Maintenance Point
LM	Loss Measurement
OAM	Operations, Administration and Maintenance
OWDM	One-Way Delay Measurement
OWLM	One-Way Loss Measurement
PDR	Packet Delivery Rate
PM	Performance Monitoring
TLV	Type, Length and Value
TRILL	Transparent Interconnection of Lots of Links
TWDM	Two-Way Delay Measurement
TWLM	Two-Way Loss Measurement

3. Loss and Delay Measurement in the TRILL Architecture

As described in [OAM-FRAMEWK], OAM protocols in a TRILL campus are used by two types of Maintenance Points (MPs); Maintenance End Points (MEPs) and Maintenance Intermediate Points (MIPs).

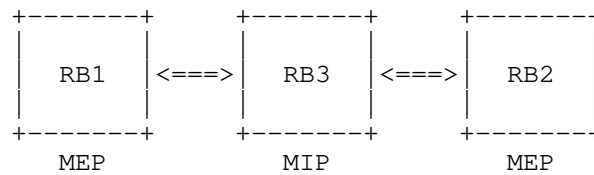


Figure 1 Maintenance Points in a TRILL Campus

Performance Monitoring (PM) allows a MEP to perform loss and delay measurements to any MEP in the campus. Performance monitoring is performed in the context of a specific Maintenance Domain (MD).

A MEP MUST support generation of PM messages, response to PM messages and computation of the packet loss and packet delay.

The PM functionality defined in this document is not applicable to MIPs.

3.1. Performance Monitoring Granularity

As defined in [OAM-FRAMEWK], PM can be applied at three levels of granularity, 'Network', 'Service' and 'Flow':

- o Network-level PM: the PM protocol is run over a dedicated test VLAN or FGL.
- o Service-level PM: the PM protocol is used to perform measurements of actual user VLANs or FGL.
- o Flow-level PM: the PM protocol is used to perform measurements on a per-flow basis. A flow, as defined in [OAM-REQ], is a set of packets that share the same path and per-hop behavior (such as priority).
As defined in [OAM-FRAMEWK], flow-based monitoring uses a Flow Entropy field that resides at the beginning of the OAM packet header (see Section 6.1.), and mimics the forwarding behavior of the monitored flow.

3.2. One-Way vs. Two-Way Performance Monitoring

Paths in a TRILL network are not necessarily symmetric, i.e., a packet sent from RB1 to RB2 does not necessarily traverse the same set of RBridges as a packet sent from RB2 to RB1. Even within a given flow, packets from RB1 to RB2 do not necessarily traverse the same path as packets from RB2 to RB1. Therefore, this document provides

tools for one-way performance monitoring and for two-way performance monitoring.

3.2.1. One-Way Performance Monitoring

In one-way PM, RB1 sends PM messages to RB2, allowing RB2 to monitor the performance on the path from RB1 to RB2.

A MEP SHOULD support one-way performance monitoring. A MEP SHOULD support both the functionality of the sender, RB1, and the functionality of the receiver, RB2.

One-way PM can be applied either proactively or on-demand, although the more typical scenario is the proactive mode, where RB1 and RB2 periodically transmit PM messages to each other, allowing each of them to monitor the performance on the incoming path from the peer MEP.

3.2.2. Two-Way Performance Monitoring

In two-way PM, a sender, RB1, sends PM messages to a reflector, RB2, and RB2 responds to these messages, allowing RB1 to monitor the performance of:

- o The path from RB1 to RB2.
- o The path from RB2 to RB1.
- o The two-way path from RB1 to RB2, and back to RB1.

Note that in some cases it may be interesting for RB1 to monitor only the path from RB1 to RB2. Two-way PM allows the sender, RB1, to monitor the path from RB1 to RB2, as opposed to one-way PM (Section 3.2.1.), which allows the receiver, RB2, to monitor this path.

A MEP MUST support two-way PM. A MEP MUST support both the sender and the reflector functionality.

As described in Section 3.1., flow-based PM uses the Flow Entropy field as one of the parameters that identify a flow. In two-way PM, the Flow Entropy of the path from RB1 to RB2 is typically different than the Flow Entropy of the path from RB2 to RB1. This document defines a Reflector Entropy TLV (Section 6.4.), which allows the sender to specify the Flow Entropy value to be used in the response message.

Two-way PM can be applied either proactively or on-demand.

3.3. Point-to-point PM vs. Point-to-multipoint PM

PM can be applied either as a point-to-point measurement protocol, or as a point-to-multi-point measurement protocol.

The point-to-point approach measures the performance between two RBridges using unicast PM messages.

In the point-to-multipoint approach an RBridge RB1 sends PM messages to multiple RBridges using multicast messages. The reflectors (in two-way PM) respond to RB1 using unicast messages.

4. Loss Measurement

The LM protocol has two flavors, One-Way Loss Measurement (OWLM), and Two-Way Loss Measurement (TWLM).

Notes: [Y.1731] defines two-way LM, but does not support one-way LM. The terms 'one-way' and 'two-way' LM should not be confused with the terms 'single-ended' and 'dual-ended' LM used in [Y.1731]. As defined in Section 3.2., the terms 'one-way' and 'two-way' specify whether the protocol monitors performance on one direction, or on both directions. The terms 'single-ended' and 'dual-ended', on the other hand, describe whether the protocol is asymmetric or symmetric, respectively.

4.1. One-Way Loss Measurement (OWLM)

OWLM measures the one-way packet loss rate from one MEP to another. The loss rate is measured using a set of One-way Synthetic Loss Measurement (1SLM) messages. The packet format of the 1SLM message is specified in Section 6.2.2. Figure 2 illustrates an OWLM message exchange.

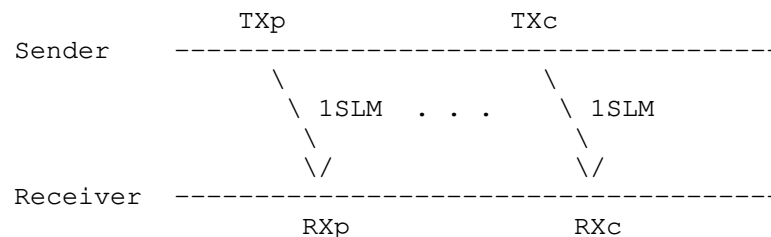


Figure 2 One-Way Loss Measurement

The OWLM procedure uses a set of 1SLM messages to measure the packet loss rate. The figure shows two non-consecutive messages from the set.

The sender maintains a counter of transmitted 1SLM messages, and includes the value of this counter, TX, in each 1SLM message it transmits. The receiver maintains a counter of received 1SLM messages, RX, and can calculate the loss rate by comparing its counter values to the counter values received in the 1SLM messages.

In Figure 2, the subscript 'c' is short for current, and 'p' is short for previous.

4.1.1. 1SLM Message Transmission

OWLM can be applied either proactively or on-demand, although as mentioned in Section 3.2.1., it is more likely to be applied proactively.

The term 'on-demand' in the context of OWLM implies that the sender transmits a fixed set of 1SLM messages, allowing the receiver to perform the measurement based on this set.

A MEP that supports OWLM MUST support unicast transmission of 1SLM messages.

A MEP that supports OWLM MAY support multicast transmission of 1SLM messages.

The sender MUST maintain a packet counter for each peer MEP and test ID. Every time the sender transmits a 1SLM packet it increments the corresponding counter, and then integrates the value of the counter into the <Counter TX> field of the 1SLM packet.

The 1SLM message MAY be sent with a Data TLV, allowing loss measurement for various packet sizes.

4.1.2. 1SLM Message Reception

The receiver MUST maintain a reception counter for each peer MEP and test ID. Upon receiving a 1SLM packet, the receiver MUST verify that:

- o The 1SLM packet is destined to the current MEP.
- o The packet's MD level matches the MEP's MD level.

If both conditions are satisfied, the receiver increments the corresponding packet counter, and records the new value of the counter, RX1.

A MEP that supports OWLM MUST support reception of both unicast and multicast 1SLM messages.

The receiver computes the one-way packet loss with respect to a measurement interval. A measurement interval includes a sequence of 1SLM message. The one-way packet loss is computed by comparing the counter values TXp and RXp at the beginning of the measurement interval, and the counter values TXc and RXc at the end of the measurement interval (Figure 2):

$$\text{one-way packet loss} = (\text{TXc}-\text{TXp}) - (\text{RXc}-\text{RXp}) \quad (1)$$

The calculation in Equation (1) is based on counter value differences, implying that the sender's counter, TX, and the receiver's counter, RX, are not required to be synchronized with respect to a common init value.

When the receiver calculates the packet loss per Equation (1) it MUST perform a wraparound check. If the receiver detects that one of the counters has wrapped around, the receiver adjusts the result of Equation (1) accordingly.

A 1SLM receiver MUST support reception of 1SLM messages with a Data TLV.

4.2. Two-Way Loss Measurement (TWLM)

TWLM allows a MEP to measure the packet loss on the paths to and from a peer MEP. TWLM uses a set of Synthetic loss Measurement Messages (SLM) to compute the packet loss. Each SLM is answered with a Synthetic loss Measurement Reply (SLR). The packet formats of the SLM and SLR packets are specified in Sections 6.2.3. and 6.2.4., respectively. Figure 2 illustrates a TWLM message exchange.

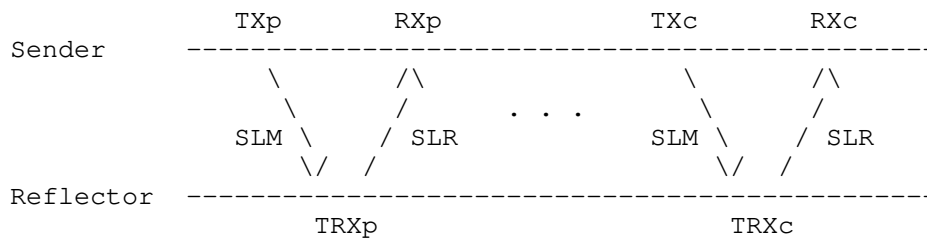


Figure 3 Two-Way Loss Measurement

The TWLM procedure uses a set of SLM-SLR handshakes. The figure shows two non-consecutive handshakes from the set.

The sender maintains a counter of transmitted SLM messages, and includes the value of this counter, TX, in each transmitted SLM message. The reflector maintains a counter of received SLM messages, TRX. The reflector generates an SLR, and incorporates TRX into the SLR packet. The sender maintains a counter of received SLR messages, RX. Upon receiving an SLR message, the sender can calculate the loss rate by comparing the local counter values to the counter values received in the SLR messages.

The subscript 'c' is short for current, and 'p' is short for previous.

4.2.1. SLM Message Transmission

TWLM can be applied either proactively or on-demand.

A MEP that supports TWLM MUST support unicast transmission of SLM messages.

A MEP that supports TWLM MAY support multicast transmission of SLM messages.

The sender MUST maintain a counter of transmitted SLM packets for each peer MEP and test ID. Every time the sender transmits an SLM packet it increments the corresponding counter, and then integrates the value of the counter into the <Counter TX> field of the SLM packet.

A sender MAY include a Reflector Entropy TLV in an SLM message. The Reflector Entropy TLV format is specified in Section 6.4.

An SLM message MAY be sent with a Data TLV, allowing loss measurement for various packet sizes.

4.2.2. SLM Message Reception

The reflector MUST maintain a reception counter, TRX, for each peer MEP and test ID.

Upon receiving an SLM packet, the reflector MUST verify that:

- o The SLM packet is destined to the current MEP.
- o The packet's MD level matches the MEP's MD level.

If both conditions are satisfied, the reflector increments the corresponding packet counter, and records the value of the new counter, TRX. The reflector then generates an SLR message that is identical to the received SLM, except for the following modifications:

- o The reflector incorporates TRX into the <Counter TRX> field of the SLR.
- o The <OpCode> field in the OAM header is set to the SLR OpCode.
- o The reflector assigns its MEP ID in the <Reflector MEP ID> field.
- o If the received SLM includes a Reflector Entropy TLV (see Section 6.4.), the reflector copies the value of the Flow Entropy from the TLV into the <Flow Entropy> field of the SLR message. The outgoing SLR message does not include a Reflector Entropy TLV.
- o The TRILL header and transport header are modified to reflect the source and destination of the SLR packet. The SLR is always a unicast message.

A MEP that supports TWLM MUST support reception of both unicast and multicast SLM messages.

A reflector MUST support reception of SLM packets with a Data TLV. When receiving an SLM with a Data TLV, the reflector includes the unmodified TLV in the SLR.

4.2.3. SLR Message Reception

The sender MUST maintain a reception counter, RX, for each peer MEP and test ID.

Upon receiving an SLR message, the sender MUST verify that:

- o The SLR packet is destined to the current MEP.
- o The <Sender MEP ID> field in the SLR packet matches the current MEP.
- o The packet's MD level matches the MEP's MD level.

If the conditions above are met, the sender increments the corresponding reception counter, and records the new value, RX.

The receiver computes the one-way packet delay with respect to a measurement interval. A measurement interval includes a sequence of 1SLM message. The one-way packet delay is performed by comparing the counter values TXp and RXp at the beginning of the measurement interval, and the counter values TXc and RXc at the end of the measurement interval (Figure 2):

The sender computes the packet loss with respect to a measurement interval. A measurement interval includes a sequence of SLM messages, and their corresponding SLR messages. The packet loss rate is computed by comparing the counters at the beginning of the measurement interval, denoted with a subscript 'p', and the counters at the end of the measurement interval, denoted with a subscript 'c' (Figure 3):

$$\text{far-end packet loss} = (\text{TXc} - \text{TXp}) - (\text{TRXc} - \text{TRXp}) \quad (2)$$

$$\text{near-end packet loss} = (\text{TRXc} - \text{TRXp}) - (\text{RXc} - \text{RXp}) \quad (3)$$

The calculations in the two equations above are based on counter value differences, implying that the sender's counters, TX and RX, and the reflector's counter, TRX, are not required to be synchronized with respect to a common init value.

When the sender calculates the packet loss per Equations (2) and (3) it MUST perform a wraparound check. If the reflector detects that one of the counters has wrapped around, the reflector adjusts the result of Equations (2) and (3) accordingly.

A sender MAY choose to monitor only the far-end packet loss, i.e., perform the computation in Equation (2), and ignore the computation in Equation (3). Note that in this case the sender can run flow-based PM of the path TO the peer MEP without using the Reflector Entropy TLV.

5. Delay Measurement

The DM protocol has two flavors, One-Way Delay Measurement (OWDM), and Two-Way Delay Measurement (TWDM).

5.1. One-Way Delay Measurement (OWDM)

OWDM is used for computing the one-way packet delay from one MEP to another. The packet format used in OWDM is referred to as 1DM, and is specified in Section 6.3.2. The OWDM message exchange is illustrated in Figure 4.

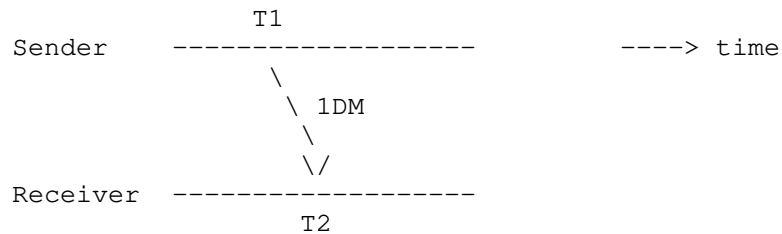


Figure 4 One-Way Delay Measurement

The sender transmits a 1DM message incorporating its time of transmission, T1. The receiver then receives the message at time T2, and calculates the one-way delay as:

$$\text{one-way delay} = T2 - T1 \quad (4)$$

Equation (4) implies that T2 and T1 are measured with respect to a common reference time. Hence, two MEPs running a OWDM protocol MUST be time-synchronized. The method used for synchronizing the two MEPs is outside the scope of this document.

5.1.1. 1DM Message Transmission

1DM packets can be transmitted proactively or on-demand, although as mentioned in Section 3.2.1., they are typically transmitted proactively.

A MEP that supports OWDM MUST support unicast transmission of 1DM messages.

A MEP that supports OWDM MAY support multicast transmission of 1DM messages.

A 1DM message MAY be sent with a Data TLV, allowing packet delay measurement for various packet sizes.

The sender incorporates the 1DM packet's time of transmission into the <Timestamp T1> field.

5.1.2. 1DM Message Reception

Upon receiving a 1DM packet, the receiver records its time of reception, T2. The receiver MUST verify two conditions:

- o The 1DM packet is destined to the current MEP.
- o The packet's MD level matches the MEP's MD level.

If both conditions are satisfied, the receiver terminates the packet and calculates the one-way delay as specified in Equation (4).

A MEP that supports OWDM MUST support reception of both unicast and multicast 1DM messages.

A 1DM receiver MUST support reception of 1DM messages with a Data TLV.

When OWDM packets are received periodically, the receiver MAY compute the packet delay variation based on multiple measurements. Note that packet delay variation can be computed even when the two peer MEPs are not time synchronized.

5.2. Two-Way Delay Measurement (TWDM)

TWDM uses a two-way handshake for computing the two-way packet delay between two MEPs. The handshake includes two packets, a Delay Measurement Message (DMM) and a Delay Measurement Reply (DMR). The

DMM and DMR packet formats are specified in Section 6.3.3. and 6.3.4., respectively.

The TWDM message exchange is illustrated in Figure 5.

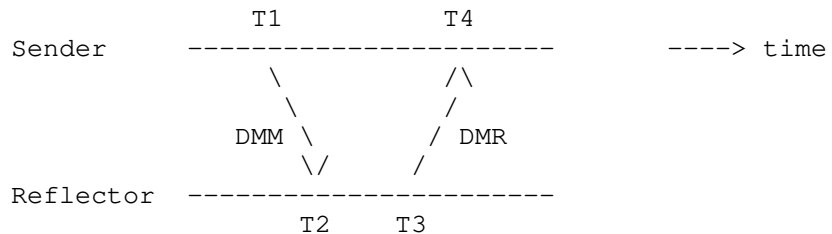


Figure 5 Two-Way Delay Measurement

The sender generates a DMM message incorporating its time of transmission, T1. The reflector receives the DMM message and records its time of reception, T2. The reflector then generates a DMR message, incorporating T1, T2 and the DMR's transmission time, T3. The sender receives the DMR message at T4, and using the 4 timestamps it calculates the two-way packet delay.

5.2.1. DMM Message Transmission

DMM packets can be transmitted periodically or on-demand.

A MEP that supports TWDM MUST support unicast transmission of DMM messages.

A MEP that supports TWDM MAY support multicast transmission of DMM messages.

A sender MAY include a Reflector Entropy TLV in a DMM message. The Reflector Entropy TLV format is specified in Section 6.4.

A DMM MAY be sent with a Data TLV, allowing packet delay measurement for various packet sizes.

The sender incorporates the DMM packet's time of transmission into the <Timestamp T1> field.

5.2.2. DMM Message Reception

Upon receiving a DMM packet, the reflector records its time of reception, T2. The reflector MUST verify two conditions:

- o The DMM packet is destined to the current MEP.
- o The packet's MD level matches the MEP's MD level.

If both conditions are satisfied, the reflector terminates the packet, and generates a DMR packet. The DMR is identical to the received DMM, except for the following modifications:

- o The reflector incorporates T2 into the <Timestamp T2> field of the DMR.
- o The reflector incorporates the DMR's transmission time, T3, into the <Timestamp T3> field of the DMR.
- o The <OpCode> field in the OAM header is set to the DMR OpCode.
- o If the received DMM includes a Reflector Entropy TLV (see Section 6.4.), the reflector copies the value of the Flow Entropy from the TLV into the <Flow Entropy> field of the DMR message. The outgoing DMR message does not include a Reflector Entropy TLV.
- o The TRILL header and transport header are modified to reflect the source and destination of the DMR packet. The DMR is always a unicast message.

A MEP that supports TWDM MUST support reception of both unicast and multicast DMM messages.

A reflector MUST support reception of DMM packets with a Data TLV. When receiving a DMM with a Data TLV, the reflector includes the unmodified TLV in the DMR.

5.2.3. DMR Message Reception

Upon receiving the DMR message, the sender records its time of reception, T4. The sender MUST verify:

- o The DMR packet is destined to the current MEP.
- o The packet's MD level matches the MEP's MD level.

If both conditions above are met, the sender uses the 4 timestamps to compute the two-way delay:

$$\text{two-way delay} = (T4-T1) - (T3-T2) \quad (5)$$

While OWDM requires the two MEPs to be synchronized, TWDM allows the sender to calculate the two-way delay without being synchronized to the reflector.

Two MEPs running a TWDM protocol MAY be time-synchronized. If TWDM is run between two time-synchronized MEPs, the sender MAY compute the one-way delays:

$$\text{one-way delay \{sender->reflector\}} = T2 - T1 \quad (6)$$

$$\text{one-way delay \{reflector->sender\}} = T4 - T3 \quad (7)$$

When TWDM is run periodically, the sender MAY also compute the delay variation based on multiple measurements.

A sender MAY choose to monitor only the sender->reflector delay, i.e., perform the computation in Equation (6), and ignore the computations in (5) and (7). Note that in this case the sender can run flow-based PM of the path TO the peer MEP without using the Reflector Entropy TLV.

6. Packet Formats

6.1. TRILL OAM Encapsulation

The TRILL OAM encapsulation is defined in [OAM-FRAMEWK], and is quoted in this document for clarity. For further details see [OAM-FRAMEWK].

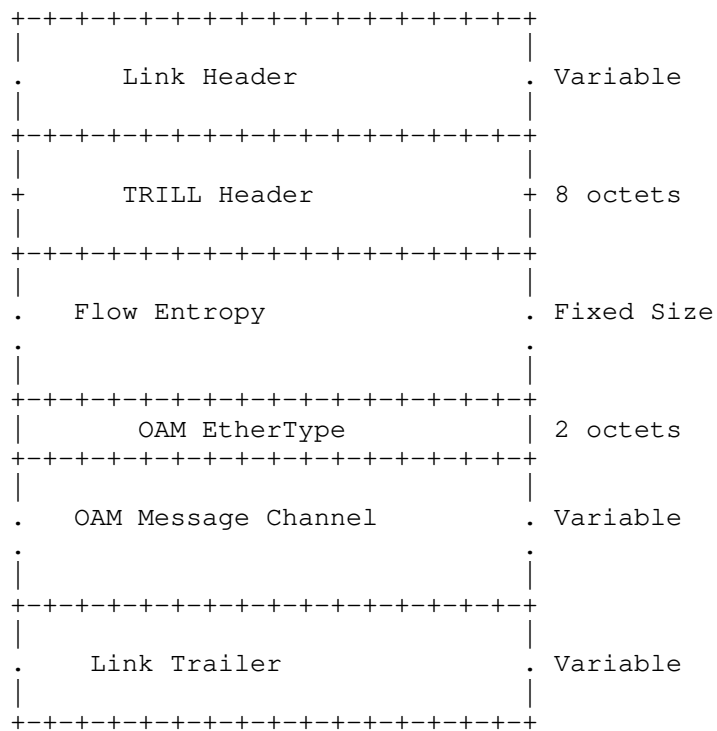


Figure 6 TRILL OAM Encapsulation

The OAM Message Channel used in this document is defined in [TRILL-FM], and has the following structure:

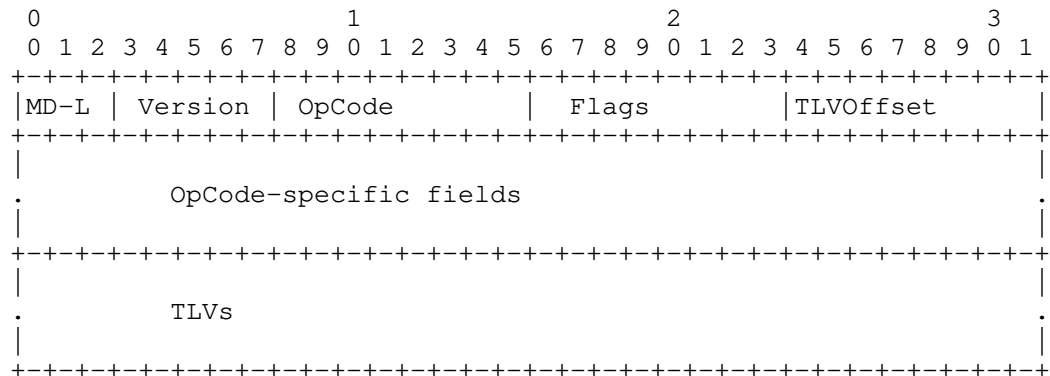


Figure 7 OAM Packet Format

The first 4 octets of the OAM Message Channel are common to all OpCodes, whereas the rest is OpCode-specific. Below is a brief summary of the fields in the first 4 octets:

- o MD-L : Maintenance Domain Level.
- o Version: indicates the version of this protocol. Always zero in the context of this document.
- o Flags: always zero in the context of this document.
- o FirstTLVOffset: defines the location of the first TLV, in octets, starting from the end of the FirstTLVOffset field.

For further details about the OAM packet format, see [TRILL-FM].

6.2. Loss Measurement Packet Formats

6.2.1. Counter Format

LM packets use a 32-bit packet counter field. When a counter is incremented beyond its maximal value, 0xFFFFFFFF, it wraps around back to 0.

6.2.2. 1SLM Packet Format

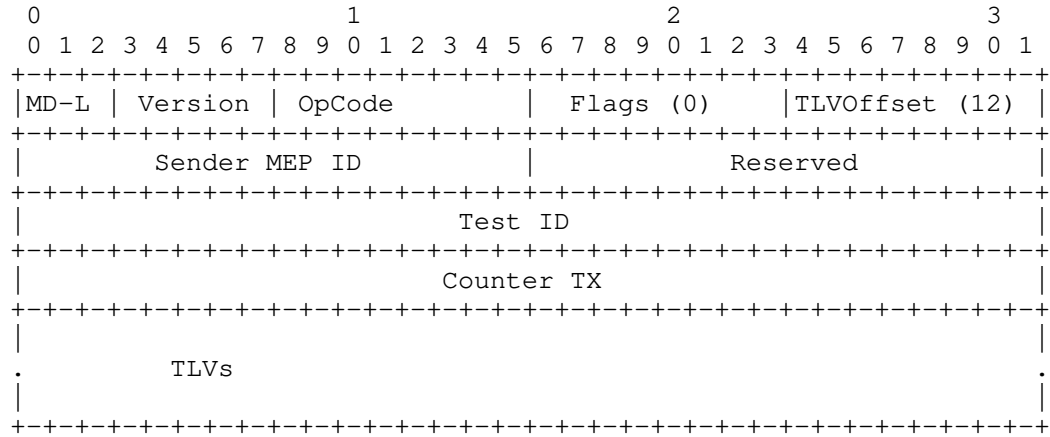


Figure 8 1SLM Packet Format

- o Sender MEP ID: the MEP ID of the MEP that initiated the 1SLM.
- o Reserved: always 0.
- o Test ID: a 32-bit unique test identifier.
- o Counter TX: the value of the sender's transmission counter, including this packet, at the time of transmission.

6.2.3. SLM Packet Format

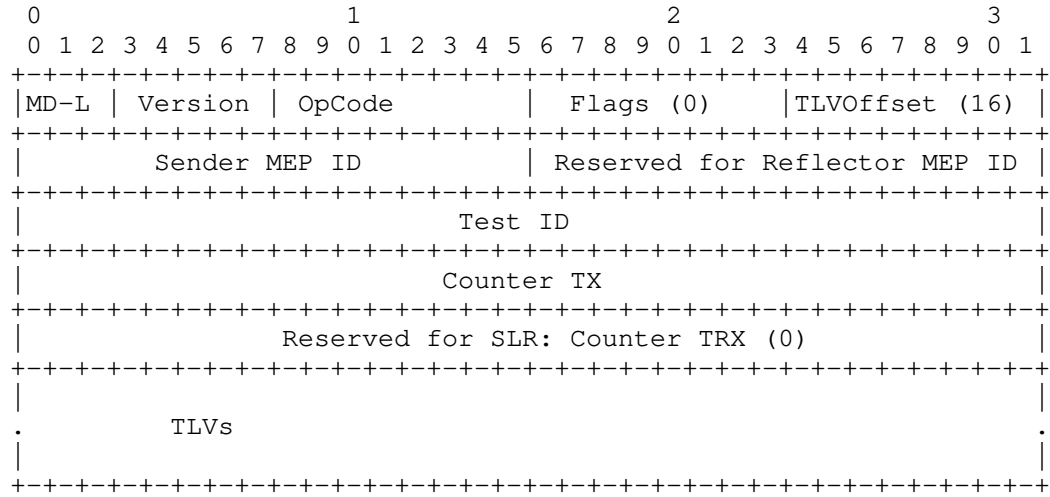


Figure 9 SLM Packet Format

- o Sender MEP ID: the MEP ID of the MEP that initiated this packet.
- o Reserved: this field is reserved for the reflector's MEP ID, to be added in the SLR.
- o Test ID: a 32-bit unique test identifier.
- o Counter TX: the value of the sender's transmission counter, including this packet, at the time of transmission.
- o Reserved: this field is reserved for the SLR corresponding to this packet. The reflector uses this field in the SLR for carrying TRX, the value of its reception counter.

6.2.4. SLR Packet Format

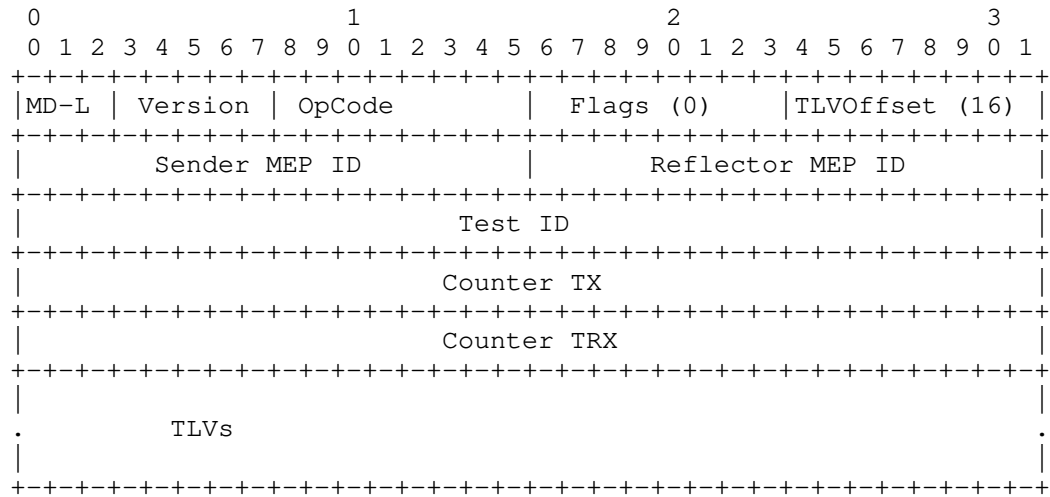


Figure 10 SLR Packet Format

- o Sender MEP ID: the MEP ID of the MEP that initiated the SLM that this SLR replies to.
- o Reflector MEP ID: the MEP ID of the MEP that transmits this SLR message.
- o Test ID: a 32-bit unique test identifier, copied from the corresponding SLM message.
- o Counter TX: the value of the sender's transmission counter at the time of the SLM transmission.
- o Counter TRX: the value of the reflector's reception counter, including this packet, at the time of reception of the corresponding SLM packet.

6.3. Delay Measurement Packet Formats

6.3.1. Timestamp Format

The timestamps used in DM packets are 64 bits long. These timestamps use the 64 least significant bits of the IEEE 1588-2008 (1588v2) Precision Time Protocol timestamp format [IEEE1588].

This truncated format consists of a 32-bit seconds field followed by a 32-bit nanoseconds field. This truncated format is also used in IEEE 1588v1, in [Y.1731], and in [MPLS-LM-DM].

6.3.2. 1DM Packet Format

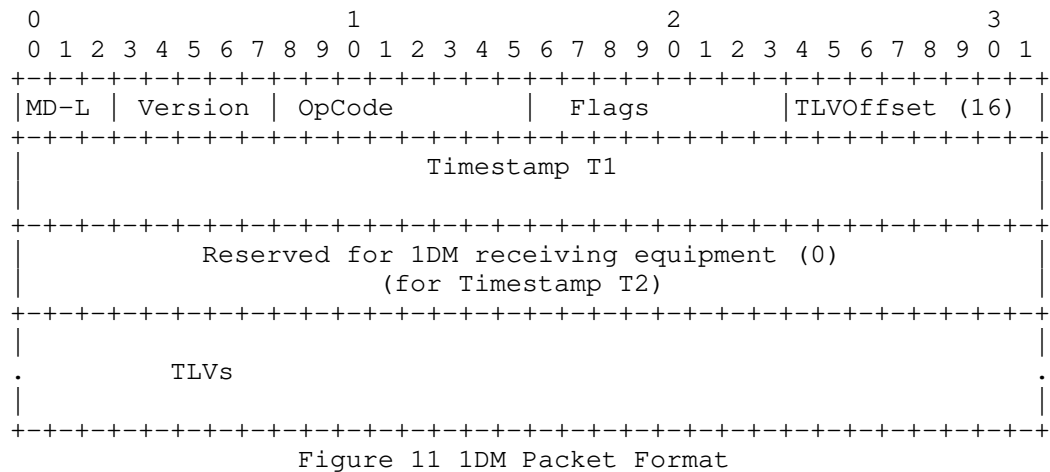


Figure 11 1DM Packet Format

- o Timestamp T1: specifies the time of transmission of this packet.
- o Reserved: this field is reserved for internal usage of the 1DM receiver. The receiver can use this field for carrying T2, the time of reception of this packet.

6.3.3. DMM Packet Format

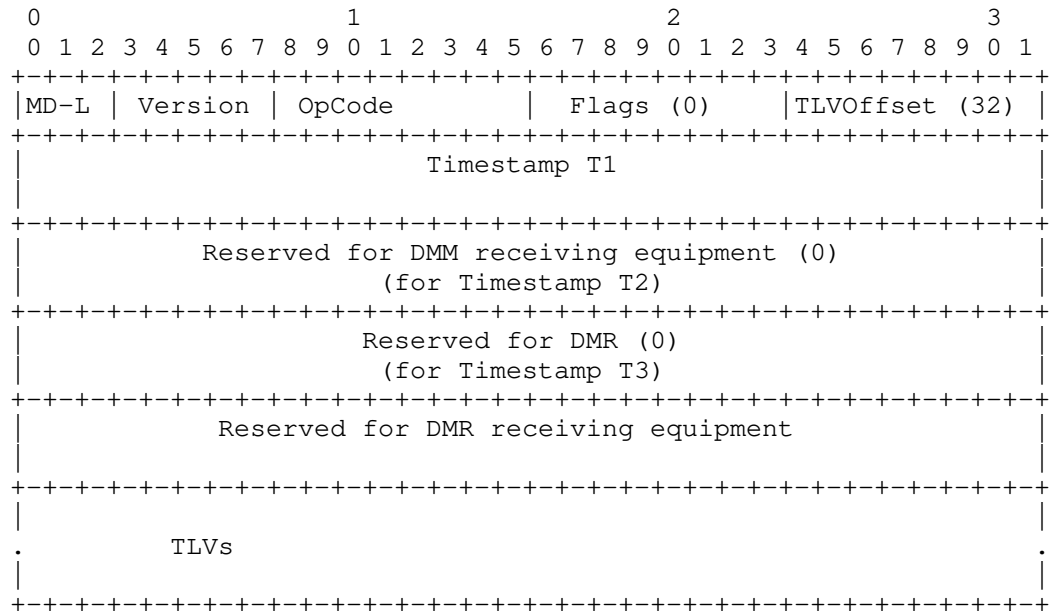


Figure 12 DMM Packet Format

- o Timestamp T1: specifies the time of transmission of this packet.
- o Reserved: this field is reserved for internal usage of the MEP that receives the DMM (the reflector). The reflector can use this field for carrying T2, the time of reception of this packet.
- o Reserved for DMR: two timestamp fields are reserved for the DMR message. One timestamp field is reserved for T3, the DMR transmission time, and the other field is reserved for internal usage of the MEP that receives the DMR.

6.3.4. DMR Packet Format

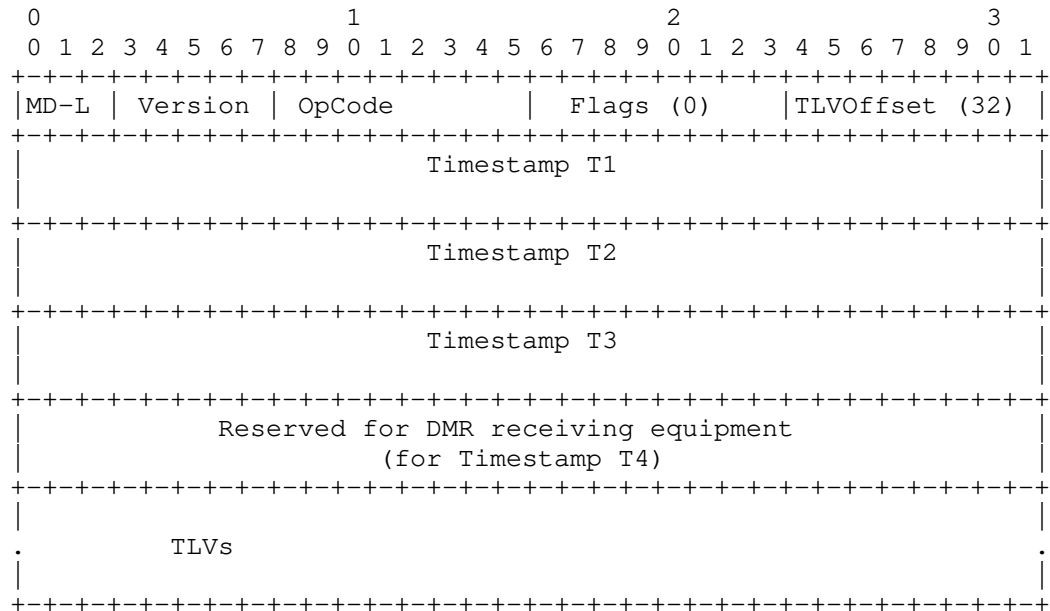


Figure 13 DMR Packet Format

- o **Timestamp T1**: specifies the time of transmission of the DMM packet that this DMR replies to.
- o **Timestamp T2**: specifies the time of reception of the DMM packet that this DMR replies to.
- o **Timestamp T3**: specifies the time of transmission of this DMR packet.
- o **Reserved**: this field is reserved for internal usage of the MEP that receives the DMR (the sender). The sender can use this field for carrying T4, the time of reception of this packet.

6.4. Reflector Entropy TLV

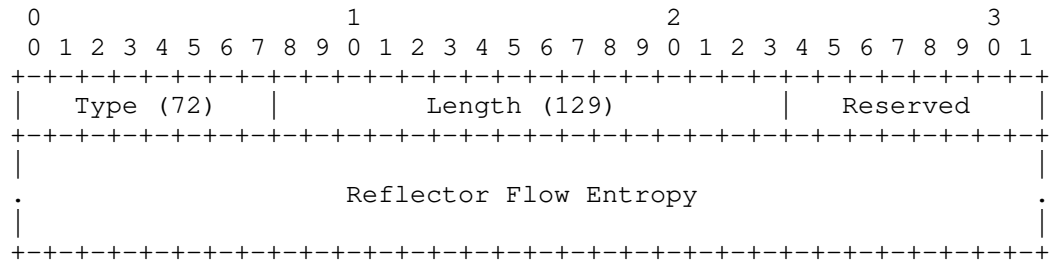


Figure 14 Reflector Entropy TLV Format

- o Type: the value 72 (see Section 8.2.) represents the Reflector Entropy TLV.
- o Length: the length of the Reflector Entropy TLV is set to 129.
- o Reserved: ignored by the recipient.
- o Reflector Flow Entropy: the 128-octet Flow Entropy to be used by the reflector.

7. Security Considerations

The security considerations of TRILL OAM are discussed in [OAM-REQ] and in [OAM-FRAMEWK]. General TRILL security considerations are discussed in [RFCTRILL]. This document does not inflict further security considerations.

8. IANA Considerations

8.1. OpCode Values

IANA is requested to assign TRILL OAM OpCode values to the packet types defined in this document. The suggested OpCode values are:

- 81 : SLM
- 80 : SLR
- 79 : 1SLM
- 83 : 1DM

85 : DMM

84 : DMR

8.2. TLV Type

IANA is requested to assign the following TLV type:

72 : Reflector Entropy TLV

9. Acknowledgments

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

10. References

10.1. Normative References

- [KEYWORDS] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2RILL] Perlman, R., Eastlake, D., Dutt, D., Gai, S., Ghanwani, A., "Routing Bridges (RBridges): Base Protocol Specification", RFC 6325, July 2011.
- [OAM-FRAMEWK] Salam, S., Senevirathne, T., Aldrin, S., Eastlake, D., "TRILL OAM Framework", draft-ietf-trill-oam-framework (work in progress), November 2012.
- [TRILL-FM] Senevirathne, T., Finn, N., Salam, S., Kumar, D., Eastlake, D., Aldrin, S., Li, Y., "TRILL Fault Management", draft-tissa-trill-oam-fm (work in progress), February 2013.

10.2. Informative References

- [OAM-REQ] Senevirathne, T., Bond, D., Aldrin, S., Li, Y., Watve, R., "Requirements for Operations, Administration and Maintenance (OAM) in TRILL (Transparent Interconnection of Lots of Links)", draft-ietf-trill-oam-req (work in progress), January 2013.
- [Y.1731] ITU-T Recommendation G.8013/Y.1731, "OAM Functions and Mechanisms for Ethernet-based Networks", July 2011.

- [802.1Q] "IEEE Standard for Local and metropolitan area networks – Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks", IEEE Std 802.1Q(tm), 2012 Edition, October 2012.
- [IEEE1588] IEEE TC 9 Instrumentation and Measurement Society, "1588 IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems Version 2", IEEE Standard, 2008.
- [MPLS-LM-DM] Frost, D., Bryant, S., "Packet Loss and Delay Measurement for MPLS Networks", RFC 6374, September 2011.
- [OAM] Andersson, L., Van Helvoort, H., Bonica, R., Romascanu, D., Mansfield, S., "Guidelines for the use of the OAM acronym in the IETF ", RFC 6291, June 2011.

Authors' Addresses

Tal Mizrahi
Marvell
6 Hamada St.
Yokneam, 20692 Israel

Email: talmi@marvell.com

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

Email: tsenevir@cisco.com

Samer Salam
Cisco
595 Burrard Street, Suite 2123
Vancouver, BC V7X 1J1, Canada

Email: ssalam@cisco.com

Donald Eastlake 3rd
Huawei USA R&D
155 Beaver Street
Milford, MA 01757 USA

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

INTERNET-DRAFT
Intended Status: Proposed Standard
Expires: July 7, 2013

Radia Perlman
Intel Labs
Fanwei Hu
ZTE Corporation
Donald Eastlake
Huawei
Kesava Vijaya Krupakaran
Dell
January 3, 2013

TRILL Smart Endnodes
draft-perlman-trill-smart-endnodes-01

Abstract

This draft addresses the problem of the size and freshness of the endnode learning table in access R Bridges, by allowing endnodes to volunteer for endnode learning and encapsulation/decapsulation. Such an endnode is known as a "smart endnode". Only the attached R Bridge can distinguish a "smart endnode" from a "normal endnode". The smart endnode uses the nickname of the attached R Bridge, so this solution does not consume extra nicknames.

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at
<http://www.ietf.org/lid-abstracts.html>

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

Copyright and License Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1	Introduction	3
1.1	Terminology	4
2.	Added information in TRILL-Hello	4
3.	Hello Exchange with RBridges	5
4.	Multi-homing	5
5.	Encapsulation and Decapsulation	6
6.	Security Considerations	8
7.	IANA Considerations	8
8.	References	8
8.1	Normative References	8
	Authors' Addresses	8

1 Introduction

The IETF TRILL (Transparent Interconnection of Lots of Links) protocol implemented by devices called RBridges (Routing Bridges, [RFC6325]), provides optimal pair-wise data frame forwarding without configuration, safe forwarding even during periods of temporary loops, and support for multipathing of both unicast and multicast traffic. TRILL accomplishes this by using IS-IS([RFC1195]) ([RFC6165]) ([RFC6326bis]) link state routing and encapsulating traffic using a header that includes a hop count. Devices that implement TRILL are called "RBridges" (Routing Bridges) or TRILL Switches.

An RBridge that attaches to endnodes is called an "edge RBridge", whereas one that exclusively forwards encapsulated frames is known as a "transit RBridge". An edge RBridge traditionally is the one that encapsulates a native Ethernet packet with a TRILL header, or that receives a TRILL-encapsulated packet and removes the TRILL header. To encapsulate, the edge RBridge must keep an "endnode table" consisting of (MAC, TRILL egress switch nickname) pairs, for those MAC addresses currently communicating with endnodes to which the edge RBridge is attached.

These table entries might be configured, received from ESADI, looked up in a directory, or learned from received traffic. If the edge RBridge has many attached endnodes, this table could become large. Also, if one of the MAC addresses in the table has moved to a different switch, it might be difficult for the edge RBridge to notice this quickly, and because the edge RBridge is tunneling to the incorrect egress RBridge, the traffic will get lost.

For these reasons, it is desirable for an endnode E (whether it be server, hypervisor, or VM) to maintain the endnode table for nodes that E is corresponding with. This eliminates the need for the attached RBridge R to know about those nodes (unless some non-smart endnode attached to R is also corresponding with those nodes), and it enables E to immediately discard an entry of (D, egress nickname), if E cannot talk to D. Then E can attempt to acquire a fresh entry for D by flooding to D, listening for ESADI, or consulting a directory.

The mechanism in this draft has E issue a TRILL-Hello (even though E is just an endnode), indicating E's desire to act as a smart endnode, together with the set of MAC addresses that E owns, and whether E would like to receive ESADI. E learns from R's Hello, whether R is capable of having a smart endnode neighbor, what R's nickname is, and which trees R can use when R ingresses frames. Although E transmits TRILL-Hellos, E does not transmit or receive LSPs.

R will accept already-encapsulated packets from E (perhaps verifying that the source MAC is indeed one of the ones that E owns, that the ingress RBridge field is R's, and if the packet is an encapsulated multideestination frame, whether the tree selected is one of the ones that R has claimed it will choose). When R receives (from the campus) a TRILL-encapsulated packet with R's nickname as egress, R checks whether the MAC address in the inner packet is one of the MAC addresses that E owns, and if so, R forwards the packet onto E's port, keeping it encapsulated.

1.1 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. Added information in TRILL-Hello

Suppose endnode E is attached to RBridge R. In order for E to act as a smart endnode, both E and R have to be signaled. The logical choice of message to do this in is a TRILL-Hello.

For smart endnode operation, R's TRILL-Hello must contain the following information:

- * flag indicating willingness to have an attached smart endnode
- * R's nickname (already included)
- * trees that R can use when ingressing frames
- * new TLV for smart endnode neighbor list
- * set of { ({set of RBridge nicknames}, pseudonode nickname) pairs}, which is a pseudonode nickname that can be used if the smart endnode is multihomed to all of the RBridge nicknames listed.

E's TRILL-Hello must contain the following information:

- * I don't want to form an RB-adjacency; merely to be a smart endnode
- * For each VLAN
 - (1) The set of MAC addresses I own
 - (2) Whether I wish to receive ESADI for that VLAN

Note that smart endnode E does not issue LSPs, nor does it receive LSPs or calculate topology. E does the following:

- o E maintains an endnode table of (MAC, nickname) of end nodes with which the smart endnode is communicating. If E is attached to multiple VLANs (traditional 12 bit VLANs or 24-bit FGL Fine Grained Labels), there would be a separate (MAC, nickname) table for each VLAN/FGL that E is attached to. Entries in this table are populated the same way that an edge RBridge populates the entries in its table:
 - * learning from (source, ingress) on packets it decapsulates
 - * from ESADI([TRILL-ESADI])
 - * by querying a directory
 - * by having some entries configured
- o When E wishes to transmit to unicast destination D, if (D, nickname) is in E's endnode table, E encapsulates with ingress nickname=R, egress nickname as indicated in D's table entry. If D is unknown, D either queries a directory or encapsulates the packet as a multdestination frame, using one of the trees that R has specified in R's TRILL-Hello.
- o When E wishes to transmit to a multicast destination, E encapsulates the packet using one of the trees that R has specified.

The attached RBridge R does the following:

- o When receiving an encapsulated frame from a port with a smart endnode, with R's nickname as ingress, R forwards the packet to the specified egress nickname, as with any encapsulated packet. However, R MAY enforce that the inner source MAC and VLAN (or FGL) are as specified for the smart endnode, by dropping if the MAC (or VLAN/FGL) are not among the expected set from the smart endnode.

3. Hello Exchange with RBridges

The smart endnode E need not send Hellos as frequently as normal RBridges. These hellos MAY be periodically unicast to the Appointed Forwarder R. In case R crashes and restarts, or the DRB changes, and E sees a Hello without mentioning E, then E SHOULD send a Hello immediately. If R is AF for any of the VLANs that E claims, R MUST list E in its Hellos as a smart endnode neighbor.

4. Multi-homing

Now suppose E is attached to the TRILL campus in two places; to RBridges R1 and R2.

There are two ways for this to work:

- (1) E can choose either R1 or R2's nickname, when encapsulating a frame, whether the encapsulated frame is sent via R1 or R2. If E wants to do active-active load splitting, and uses R1's nickname when forwarding through R1, and R2's nickname when forwarding through R2, this will cause distant RBridges (or smart endnodes) to keep changing their endnode table entry for D between (D, R1's nickname) and (D, R2's nickname). So it would be preferable for E to always encapsulate using the same nickname (R1 or R2) unless E detects a problem with connectivity using that nickname. And in this case, R1 and R2 need to be informed that the smart endnode might encapsulate with a different nickname, i.e., R1 might receive an encapsulated packet from smart endnode E using ingress nickname "R2".
- (2) R1 and R2 might indicate, in their Hello, another nickname that attached end nodes may use if they are multihomed to R1 and R2, separate from R1 and R2's nicknames (which they would also list in their Hello). This would be useful if there were many end nodes multihomed to the same set of RBridges. This would be analogous to a pseudonode nickname; return traffic would go via the shortest path from the source to the endnode, whether it is R1 or R2. If E loses connectivity to R2, then E would revert to using R1's nickname. This does use a nickname, but hopefully would be shared by many end nodes multihomed to the same set of RBridges.

5. Encapsulation and Decapsulation

Consider a smart endnode E on a shared LAN wishing to communicate with D. First suppose D is not on the shared LAN. The draft already handles that case.

Suppose D is on the same shared LAN as smart node E. If E does not know where D is, the packet needs to be flooded BOTH on the shared LAN as a native packet, and throughout the campus, encapsulated.

- (1) If E does not know where D is, then E sends two copies of the packet; one native, and one encapsulated.
- (2) If the Appointed Forwarder R receives a native packet on a port with smart endnode E, and the source MAC is one that E owns, then

R MUST discard the packet.

- (3) If R receives a native packet on a port with smart endnode E, and the destination MAC is one that E owns, then R MUST discard the packet.
- (4) The other non-AFs in the shared LAN behave as usual - they don't encapsulate native frames.

This solution works regardless of whether D is a smart endnode or not. Smart endnode E will learn that D is on the shared link, and keep in its table (D, native on my link). So in the future, E will send to D by transmitting natively. R MUST discard the packet because it notices the source MAC is owned by E. D will transmit to E natively, whether or not D is a smart endnode. R will also discard the packet in this case because the destination MAC is owned by E. So D and E will talk natively.

If R receives a multicast from a remote RBridge, and the exit interface includes hybrid endnodes, it should send two copies of mulicast frames, one as native and the other as TRILL encapsulated frame. When smart endnode receives the encapsulated frame, it learns the remote address.

6. Security Considerations

For general TRILL Security Considerations, see([RFC6325]).

7. IANA Considerations

This document requires no IANA actions.

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.
- [RFC1195] Callon, R., "Use of OSI IS-IS for routing in TCP/IP and dual environments", RFC 1195, December 1990.
- [RFC6325] R. Perlman, D. Eastlake, et al, "RBridges: Base Protocol Specification", RFC 6325, July 2011.
- [RFC6165] Banerjee, A. and D. Ward, "Extensions to IS-IS for Layer-2 Systems", RFC 6165, April 2011.
- [RFC6326bis] D. Eastlake, A. Banerjee, et al, "Transparent Interconnection of Lots of Links (TRILL) Use of IS-IS", draft-eastlake-isis-rfc6326bis-09.txt, work in progress.
- [Directory] Linda, D., Eastlake, D., Perlman, R., and I. Gashinsky, "TRILL Edge Directory Assistance Framework", trill-directory-framework-01 (work in process).
- [TRILL-ESADI] Zhai, H., Hu, F., Perlman, R., and D. Eastlake, "TRILL(Transparent Interconnection of Lots of Links): The ESADI (End Station Address Distribution Information) Protocol", draft-ietf-trill-esadi-01(work in process).

Authors' Addresses

Radia Perlman
Intel Labs
2200 Mission College Blvd.
Santa Clara, CA 95054-1549 USA

Phone: +1-408-765-8080

Email: Radia@alum.mit.edu

Fangwei Hu
ZTE Corporation
No.889 Bibo Rd
Shanghai, 201203
China

Phone: +86 21 68896273
Email: hu.fangwei@zte.com.cn

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

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

Kesava Vijaya Krupakaran
Dell
Olympia Technology Park,
Guindy Chennai 600 032
India

Phone: +91 44 4220 8496
Email: Kesava_Vijaya_Krupak@Dell.com

TRILL Working Group
Internet Draft
Intended status: Standard Track

Tissa Senevirathne
Norman Finn
Samer Salam
Deepak Kumar
CISCO

Donald Eastlake
Sam Aldrin
Yizhou Li
Huawei

February 17, 2013

Expires: August 2013

TRILL Fault Management
draft-tissa-trill-oam-fm-01.txt

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

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

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

This Internet-Draft will expire on August 17, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Abstract

TRILL OAM Fault Management solution is presented in this document. Methods proposed in this document follow the IEEE 802.1 CFM framework and reuse OAM tools where possible. Additional messages and TLVs are defined for TRILL specific applications or where different set of information is required than IEEE 802.1 CFM.

Table of Contents

1. Introduction.....	4
2. Conventions used in this document.....	4
3. General Format of TRILL OAM frames.....	5
3.1. Identification of TRILL OAM frames.....	7
3.2. Use of TRILL OAM Flag.....	7
3.2.1. Handling of TRILL frames with "A" Flag.....	8
3.3. Backwards Compatibility Method.....	8
3.4. OAM Capability Announcement.....	9
4. TRILL OAM Layering vs. IEEE Layering.....	10
4.1. Processing at ISS Layer.....	11
4.1.1. Receive Processing.....	11
4.1.2. Transmit Processing.....	11
4.2. End Station VLAN and Priority Processing.....	11
4.2.1. Receive Processing.....	11
4.2.2. Transmit Processing.....	11
4.3. TRILL Encapsulation and De-capsulation Layer.....	11
4.3.1. Receive Processing for Unicast packets.....	11
4.3.2. Transmit Processing for unicast packets.....	12
4.3.3. Receive Processing for Multicast packets.....	12
4.3.4. Transmit Processing of Multicast packets.....	13
4.4. TRILL OAM Layer Processing.....	14
5. Maintenance Associations (MA) in TRILL.....	15
6. MEP Addressing.....	16
6.1. Use of MIP in TRILL.....	19
7. Approach for Backwards Compatibility.....	21
8. Continuity Check Message (CCM).....	22
9. TRILL OAM Message Channel.....	24
9.1. TRILL OAM Message header.....	24

9.2. TRILL OAM Opcodes.....	25
9.3. Format of TRILL OAM TLV.....	25
9.4. TRILL OAM TLVs.....	26
9.4.1. Common TLVs between 802.1ag and TRILL.....	26
9.4.2. TRILL OAM Specific TLVs.....	26
9.4.2.1. TRILL OAM Application Identifier TLV.....	27
9.4.3. Out Of Band Reply Address TLV.....	28
9.4.3.1. Diagnostics Label TLV.....	29
9.4.3.2. Original Data Payload TLV.....	30
9.4.3.3. RBridge scope TLV.....	30
9.4.3.4. Previous RBridge nickname TLV.....	31
9.4.3.5. Next Hop RBridge List TLV.....	31
9.4.3.6. Multicast Receiver Port count TLV.....	32
9.4.4. Flow Identifier (flow-id) TLV.....	33
10. Loopback Message.....	34
10.1.1. Loopback OAM Message format.....	34
10.1.2. Theory of Operation.....	34
10.1.2.1. Originator RBridge.....	34
10.1.2.2. Intermediate RBridge.....	35
10.1.2.3. Destination RBridge.....	35
11. Path Trace Message.....	36
11.1.1. Theory of Operation.....	36
11.1.1.1. Originator RBridge.....	36
11.1.1.2. Intermediate RBridge.....	37
11.1.1.3. Destination RBridge.....	38
12. Multi-Destination Tree Verification (MTV) Message.....	38
12.1. Multi-Destination Tree Verification (MTV) OAM Message Format.....	39
12.2. Theory of Operation.....	39
12.2.1. Originator RBridge.....	39
12.2.2. Receiving RBridge.....	40
12.2.3. In scope RBridges.....	40
13. Application of Continuity Check Message (CCM) in TRILL.....	41
13.1. CCM Error notification - Method-1.....	42
13.2. CCM Error Notification Method-2.....	43
13.3. Theory of Operation.....	44
13.3.1. Originator RBridge.....	44
13.3.2. Intermediate RBridge.....	45
13.3.3. Destination RBridge.....	45
14. Multiple Fragment Reply.....	45
15. Security Considerations.....	46
16. Allocation Considerations.....	46
16.1. IEEE Allocation Considerations.....	46
16.2. IANA Considerations.....	46
17. References.....	47
17.1. Normative References.....	47
17.2. Informative References.....	47
18. Acknowledgments.....	48

1. Introduction

The general structure of TRILL OAM messages is presented in [TRILLOAMFM]. According to [TRILLOAMFM], TRILL OAM messages consist of five parts: link header, TRILL header, flow entropy, OAM message channel, and link trailer.

The OAM message channel allows defining various control information and carrying OAM related data between TRILL switches, also known as R Bridges or Routing Bridges.

The OAM message channel, if defined properly, can be shared between different technologies. A common OAM channel allows a uniform user experience for the customers, savings on operator training, re-use of software code base and faster time to market.

This document uses the message format defined in IEEE 802.1ag Connectivity Fault Management (CFM) [8021Q] as the basis for the TRILL OAM message channel.

The ITU-T Y.1731 standard utilizes the same messaging format as [8021Q] and OAM messages where applicable. In this document, we take a similar stance and propose reusing [8021Q] in TRILL OAM. We assume readers are familiar with [8021Q] and Y1731. Readers who are not familiar with these documents are encouraged to review [8021Q] and Y1731.

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].

Acronyms used in the document include the following:

MP - Maintenance Point [TRILLOAMFM]

MEP - Maintenance End Point [TRILLOAMFM] [8021Q]

MIP - Maintenance Intermediate Point [TRILLOAMFM] [8021Q]

MA - Maintenance Association [8021Q] [TRILLOAMFM]

CCM - Continuity Check Message [8021Q]

LBM - Loop Back Message [8021Q]

PTM - Path Trace Message

MTV - Multi-destination Tree Verification Message

OAM - Operations, Administration, and Maintenance [RFC6291]

TRILL - Transparent Interconnection of Lots of Links [RFC6325]

FGL - Fine Grained Label [RFC6325]

3. General Format of TRILL OAM frames

The TRILL forwarding paradigm allows an implementation to select a path from a set of equal cost paths to forward a packet. Selection of the path of choice is implementation dependent. However, it is a common practice to utilize Layer 2 through Layer 4 information in the frame payload for path selection.

For accurate monitoring and/or diagnostics, OAM Messages are required to follow the same path as corresponding data packets. [TRILLOAMFM] proposes a high-level format of the OAM messages. The details of the TRILL OAM frame format are defined in this document.

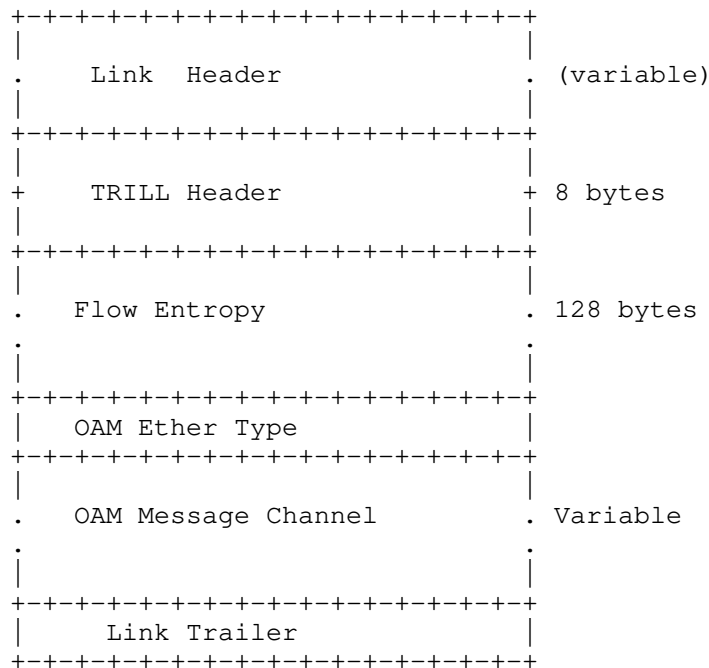


Figure 1 Format of TRILL OAM Messages

Link Header: Media-dependent header. For Ethernet, this includes Destination MAC, Source MAC, VLAN (optional) and EtherType fields.

TRILL Header: Minimum of 8 bytes when the Extended Header is not included [RFC6325]

Flow Entropy: This is a 128-byte fixed size opaque field. The least significant bits of the field **MUST** be padded with zeros, up to 128 bytes, when the flow entropy is less than 128 bytes. Flow entropy enables emulation of the forwarding behavior of the desired data packets.

OAM Ether Type: OAM Ether Type is 16-bit EtherType that identifies the OAM Message channel which follows. This document specifies using the EtherType allocated for 802.1ag for this purpose. Identifying the OAM Message Channel with a dedicated EtherType allows the easy identification of the beginning of the OAM message channel across multiple standards.

OAM Message Channel: This is a variable size section that carries OAM related information. We propose reusing the message format defined in [8021Q] for this purpose.

Link Trailer: Media-dependent trailer. For Ethernet, this is the FCS (Frame Check Sequence).

3.1. Identification of TRILL OAM frames

TRILL, as originally specified in [RFC6325], did not have a specific flag or a method to identify OAM frames. This document updates RFC6325 to include specific methods to identify TRILL OAM frames. Section 3.2. below explains the details of the method. However, it is important, for backwards compatibility reasons, to define methods of identifying TRILL OAM frames without using these extensions. Section 3.3. presents a set of possible methods for identifying OAM frames without using the proposed extensions of section 3.2. The methods defined in section 3.3. impose limitations on the construction of the flow entropy field of the OAM frames and SHOULD be used for backwards compatibility scenarios only.

3.2. Use of TRILL OAM Flag

The TRILL Header, as defined in [RFC6325], has two reserved bits that are currently unused. R Bridges are currently required to ignore these fields. This document specifies use of the reserved bit next to Version field in the TRILL header as the Alert flag. Alert flag will be denoted by 'A'. (TISSA: Move to A)

Implementations that follow the extension of using the "A" flag to identify frames MUST exclusively use that flag and methods specified in section 3.2.1. The "A" flag MUST NOT be utilized for forwarding decisions such as the selection of ECMP paths, etc.

```

+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| V | A | R | M | Op-Length | Hop Count |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   Egress RBridge Nickname   |   Ingress RBridge Nickname   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   Options...               |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Figure 2 TRILL Header

A (1 bit) - Indicates this is a possible OAM frame and is subject to specific handling as specified in this document.

All other fields carry the same meaning as defined in RFC6325.

3.2.1. Handling of TRILL frames with "A" Flag

Value "1" in the A flag indicates TRILL frames that may qualify as OAM frames. Implementations are further required to validate such frames by comparing the value at the OAM Ether Type (Figure 1) location with the CFM EtherType "0x8902" [8021Q]. If the value matches, such frames are identified as TRILL OAM frames and SHOULD be processed as discussed in Section 4.

3.3. Backwards Compatibility Method

For unicast frames, TRILL OAM packets are identified by its TRILL egress nickname and the presence of either Reserved Inner.MacSA (TBD) or OAM Ether Type 0x8902 [8021Q].

For multicast frames, TRILL OAM packets are identified by either OAM EtherType 0x8902 [8021Q] or Reserved Inner.MacSA (TBD) .

The following table summarizes the identification of different OAM frames from data frames.

Flow Entropy	Inner MacSA	OAM Ether Type	Egress nickname
unicast L2	N/A	Match	Match
Multicast L2	N/A	Match	N/A
Unicast IP	Match	N/A	Match
Multicast IP	Match	N/A	N/A
Notification	N/A	Match	Match

Figure 3 Identification of TRILL OAM Frames

3.4. OAM Capability Announcement

Any given TRILL RBridge can be one of: OAM incapable OR OAM capable with new extensions OR OAM capable with backwards-compatible method. The OAM request originator, prior to origination of the request is required to identify the OAM capability of the target and generate the appropriate OAM message.

We propose to utilize the capability flags defined in TRILL version sub-TLV (TRILL-VER) [rfc6326bis]. The following Flags are defined:

O - OAM Capable

B - Backwards Compatible.

A capability announcement, with O Flag set to 1 and B flag set to 1, indicates that the implementation is OAM capable but utilize backwards compatible method defined in section 3.3. A capability announcement, with O Flag set to 1 and B flag set to 0, indicates that the implementation is OAM capable and utilizes the method specified in section 3.2.

When O Flag is set to 0, the announcing implementation is considered not capable of OAM and B flag is ignored on the receiving side.

```

+-----+
| Type                | (1 byte)
+-----+
| Length              | (1 byte)
+-----+
| Max-version         | (1 byte)
+-----+
| A|O|B|Other Capabilities and Header Flags| (4 bytes)
+-----+
0                               1                               3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 0 1

```

Figure 4 TRILL-VER sub-TLV [rfc6326bis] with O and B flags

NOTE: Bit position of O and B flags in the TRILL-VER sub-TLV are presented above as an example. Actual positions of the flags will be determined by TRILL WG and IANA and future revision of this document will be updated to include the allocations.

4. TRILL OAM Layering vs. IEEE Layering

In this section we present the placement of the TRILL OAM shim within the IEEE 802.1 layers. The processing of both the Transmit and Receive directions is explained.

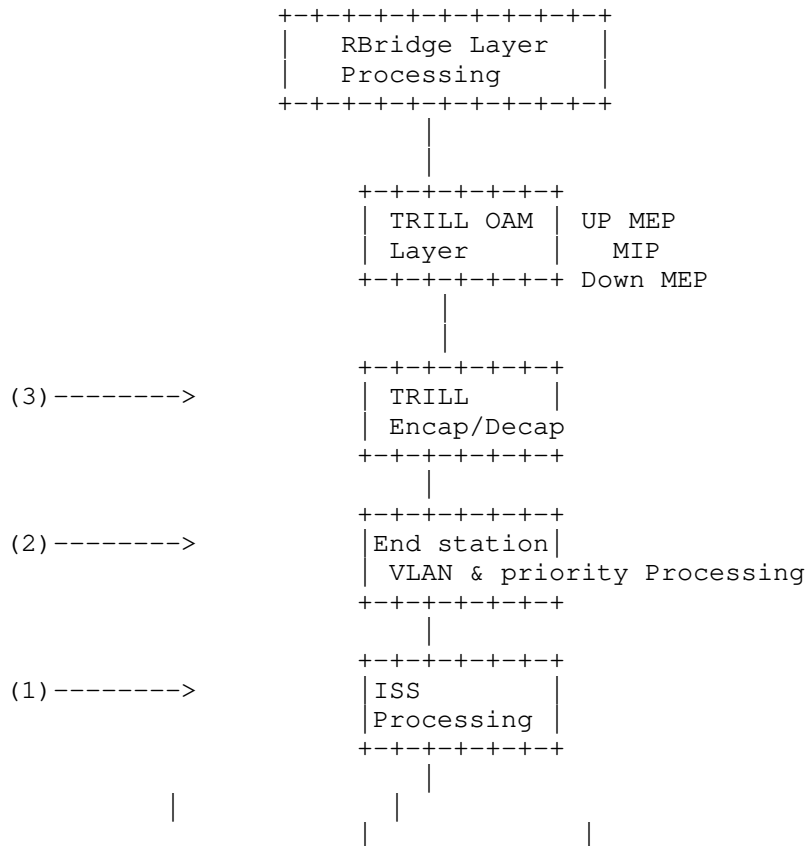


Figure 5 Placement of TRILL MP within IEEE 802.1

[RFC6325] Section 4.6 provides a detail explanation of frame processing. Please refer to [RFC6325] for processing scenarios not covered herein.

4.1. Processing at ISS Layer

4.1.1. Receive Processing

The ISS Layer receives an indication from the port. It extracts DA, SA and marks the remainder of the payload as M1. ISS Layer passes on (DA,SA,M1) as an indication to the higher layer.

For TRILL frames, this is Outer DA and Outer SA. M1 is the remainder of the packet from the VLAN EtherType onwards.

4.1.2. Transmit Processing.

ISS layer receives indication from the higher layer that contains (DA,SA,M1). It constructs an Ethernet frame and passes down to the port.

4.2. End Station VLAN and Priority Processing

4.2.1. Receive Processing

Receives (DA,SA,M1) indication from ISS Layer. Extracts the VLAN from the M1 part of the received indication and construct (DA,SA,VLAN,PRI,M2). VLAN+PRI+M2 map to M1 in the received indication. Pass (DA,SA,VLAN,PRI,M2) to the TRILL encap/decap procession layer.

4.2.2. Transmit Procession

Receive (DA,SA,VLAN+PRI,M2) indication from TRILL encap/decap processing layer. Merge VLAN, M2 to from M1. Pass down (DA,SA,M1) to the ISS processing Layer.

4.3. TRLL Encapsulation and De-capsulation Layer

4.3.1. Receive Processing for Unicast packets

Receive indication (DA,SA,VLAN, PRI, M2) from End Station VLAN and Priority Processing Layer.

- o If DA matches Local DA and Frame is of TRILL EtherType
 - . Discard DA, SA, VLAN, PRI. From M2, derive (TRILL-HDR, iDA, iSA, i-VL, M3)
 - . If TRILL nickname is Local and TRILL-OAM Flag is set
- Pass on to OAM processing

- . Else pass on (TRILL-HDR, iDA, iSA, i-VL, M3) to RBridge Layer
- o If DA matches local DA and EtherType is not TRILL type
 - . Discard frame
- o If DA does not match and port is Appointed Forwarder and EtherType is not TRILL
 - . Insert TRILL-Hdr and send (TRILL-HDR, iDA,iSA,i-VL, M3) indication to RBridge Layer <- This is the edge function

4.3.2. Transmit Processing for unicast packets

- o Receive indication (TRILL-HDR, iDA, iSA, iVL, M3) from RBridge Layer
- o If egress TRILL nickname is local
 - o If port is Appointed Forwarder and (TRILL Alert Flag set and OAM EtherType present) then
 - . Strip TRILL-HDR and construct (DA, SA, VLAN, M2)
 - o Else
 - . Set discard flag
- o If egress TRILL nickname is not local
 - o Insert Outer DA, Outer SA, Outer VLAN, TRILL EtherType and construct (DA,SA,VLAN,M2). Where M2 is (TRILL-HDR, iDA, iSA, iVL, M)
- o Else set the discard flag
- o If discard flag is false forward (DA,SA,V,M2) to the VLAN End Station processing Layer. Otherwise, discard the packet.

4.3.3. Receive Processing for Multicast packets

- o Receive (DA,SA,V,M2) from VLAN end station processing layer
- o If the DA matches the Well-known TRILL multicast MAC address and Ethertype of the frame is TRILL
 - o Strip DA,SA and V. From M2, construct (TRILL-HDR, iDA, iSA, iVL and M3).

- o If TRILL OAM Flag is set and Ether Type OAM is present at the end of Flow entropy
 - . Perform OAM Processing
- o Else extract the TRILL header, inner MAC addresses and inner VLAN and pass indication (TRILL-HDR, iDA, iSA, iVL and M3) to TRILL RBridge Layer
- o If the DA matches the well-known TRILL multicast MAC address but EtherType is not TRILL
 - o Discard the packet
- o If the DA does not match the well-known TRILL multicast MAC address and Ether Type is not TRILL type
 - o Insert TRILL-HDR and construct (TRILL-HDR, iDA, iSA, iVL, M3)
 - o Pass the (TRILL-HDR, iDA, iSA, iVL, M3) to RBridge Layer
- o Else
 - o Discard the packet

4.3.4. Transmit Processing of Multicast packets

- o Receive indication (TRILL-HDR, iDA, iSA, iVL, M3) from RBridge layer.
 - o If TRILL-HDR multicast flag set and TRILL-HDR Alert flag set and OAM EtherType present then:
 - o (DA,SA,V,M2) by inserting TRILL ODA, OSA, O-VL and TRILL ether type. M2 here is (EtherType TRILL, TRILL-HDR, iDA, iSA, iVL, M)
- NOTE: Second copy of native format is not made.
- o Else If TRILL-HDR multicast flag set and Alert flag not set
 - o If the port is appointed Forwarder Strip TRILL-HDR, iSA, iDA, iVL and construct (DA,SA,V,M2) for native format.
 - o Make a second copy (DA,SA,V,M2) by inserting TRILL ODA, OSA, O-VL and TRILL ether type. M2 here is (EtherType TRILL, TRILL-HDR, iDA, iSA, iVL, M)

- o Else unicast packets as defined in section 4.3.2.
- o Pass the indication (DA,SA,V,M2) to End Station VLAN processing layer.

4.4. TRILL OAM Layer Processing

TRILL OAM Processing Layer is located between the TRILL Encapsulation and De-capsulation layer and RBridge Layer. It performs 1. Identification of OAM frames that need local processing
2. Perform OAM processing or redirect to the CPU for OAM processing.

- o Receive indication (TRILL-HDR, iDA, iSA, iVL, M3) from RBridge layer.
- o If the TRILL Multicast Flag is set and TRILL Alert Flag is set and TRILL OAM EtherType is present then
 - o If MEP or MIP is configured on the inner VLAN of the packet then
 - . discard packets that have MD-LEVEL Less than that of the MEP or packets that does not have MD-LEVEL present (e.g due to packet truncation).
 - . If MD-LEVEL matches MD-LEVEL of the MEP then
 - . Re-direct to OAM Processing (Do not forward further)
 - . If MD-LEVEL matches MD-LEVEL of MIP then
 - . Make a Copy for OAM processing and continue
- o Else if TRILL Alert Flag is set and TRILL OAM EtherType is present then
 - o If MEP or MIP is configured on the inner VLAN of the packet then
 - . discard packets that have MD-LEVEL not present or MD-LEVEL is Less than the that of the MEP.
 - . If MD-LEVEL matches MD-LEVEL of the MEP then
 - . Re-direct to OAM Processing (Do not forward further)
 - . If MD-LEVEL matches MD-LEVEL of MIP then
 - . Make a Copy for OAM processing and continue
- o Else // Non OAM l Packet
 - o Continue
- o Pass the indication (DA,SA,V,M2) to End Station VLAN processing layer.

NOTE: In the Received path, processing above compares against Down MEP and MIP Half functions. In the transmit processing it compares against Up MEP and MIP Half functions.

Appointed Forwarder is a Functionality that TRILL Encap/De-Cap layer performs. TRILL Encap/De-cap Layer is responsible for prevention of leaking of OAM packets as native frames.

5. Maintenance Associations (MA) in TRILL

[8021Q] defines a maintenance association as a logical relationship between a group of nodes. Each Maintenance Association (MA) is identified with a unique MAID of 48 bytes [8021Q]. CCM and other related OAM functions operate within the scope of an MA. The definition of MA is technology independent. Similarly it is encoded within the OAM message, not on the technology dependent portion of the packet. Hence we propose to utilize the MAID as defined in [8021Q]. This also allows us to utilize CCM and LBM messages defined in [8021Q], as is.

In TRILL, an MA may contain two or more RBridges (MEPs). For unicast, it is likely that the MA contains exactly two MEPs that are the two end-points of the flow. For multicast, the MA may contain two or more MEPs.

For TRILL, in addition to all of the standard 802.1Q MIB definitions, each MEP's MIB contains one or more flow entropy definitions corresponding to the set of flows that the MEP monitors.

We propose to augment the [8021Q] MIB to add the TRILL specific information. Figure 6, below depicts the augmentation of the CFM MIB to add the TRILL specific Flow Entropy.

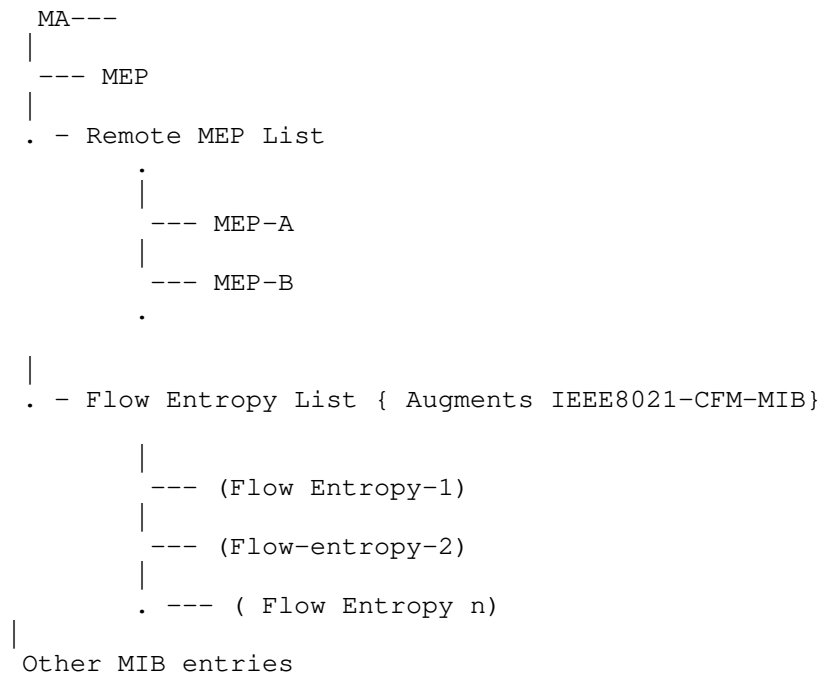


Figure 6 Correlation of TRILL augmented MIB

6. MEP Addressing

In IEEE 802.1ag [8021Q], OAM messages address the target MEP by utilizing a unique MAC address. In TRILL, for qualifying OAM packets, we propose to use a combination of the egress RBridge nickname and Inner VLAN/FGL to address the MEP.

At the MEP, OAM packets go through a hierarchy of op-code de-multiplexers. The op-code de-multiplexers channel the incoming OAM packets to the appropriate message processor (e.g. LBM) The reader may refer to Figure 7 below for a visual depiction of these different de-multiplexers.

1. Identify the packets that need OAM processing at the Local RBridge Section 4.

- a. Identify the MEP that is associated with the Inner VLAN.

2. MEP first validate the MD-LEVEL and then
 - a. Redirect to MD-LEVEL De-multiplexer
3. MD-LEVEL de-multiplexer compares the MD-Level of the packet against the MD level of the local MEPs of a given MD-Level on the port (Note: there can be more than one MEP at the same MD-Level but belonging to different MAs)
 - a. If the packet MD-LEVEL is equal to the configured MD-LEVEL of the MEP, then pass to the Opcode de-multiplexer
 - b. If the packet MD-LEVEL is less, then the configured MD-LEVEL of the MEP discard the packet
 - c. If the packer MD-LEVEL is greater, then the configured MD-LEVEL of the MEP pass on to the next higher MD-LEVEL de-multiplexer, if available. Otherwise, if no such higher MD-LEVEL de-multiplexer exists then forward the packet as normal data.
4. Opcode De-multiplexer compares the opcode in the packet with supported opcodes
 - a. If Op-code is CCM, LBM, LBR, PTM, PTR, MTVM, MTVR, then pass on to the correct Processor
 - b. If Op-code is Unknown, then discard.

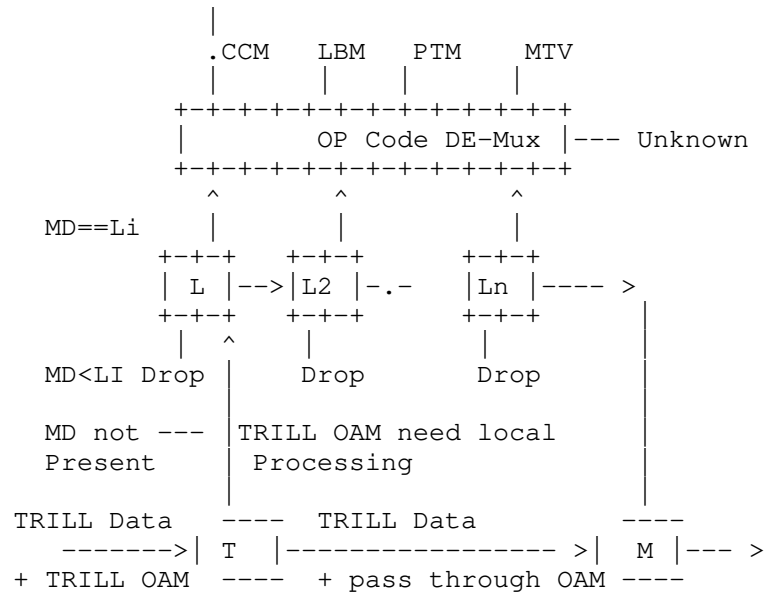


Figure 7 OAM De-Multiplexers at MEP for active SAP

T : Denotes Tap, that identifies OAM frames that need local processing. These are the packets with OAM flag set AND OAM Ether type is present after the flow entropy of the packet

M : Is the post processing merge, merges data and OAM messages that are pass through. Additionally, Merge component ensure, as explained earlier, OAM packets are not forwarded out as native frames.

L : Denotes MD-Level processing. Packets with MD-Level less than the Level will be dropped. Packets with equal MD-Level are passed on to the opcode de-multiplexer. Others are passed on to the next level MD processors or eventually to the merge point (M).

NOTE: LBM, MTV and PT are not subject to MA de-multiplexers. These packets do not have an MA encoded in the packet. Adequate response can be generated to these packets, without loss of functionality, by any of the MEP present on that interface or an entity within the RBridge.

6.1. Use of MIP in TRILL

Maintenance Intermediate Points (MIP) are mainly used for fault isolation. Link Trace Messages in [8021Q] utilize a well-known multicast MAC address and MIPs generate responses to Link Trace messages. Response to Link Trace messages or lack thereof can be used for fault isolation in TRILL.

As explained in section 11. , we propose to use a hop-count expiry approach for fault isolation and path tracing. The approach is very similar to the well-known IP trace-route approach. Hence, explicit addressing of MIPs is not required for the purpose of fault isolation.

Any given RBridge can have multiple MIPs located within a interface. As such, a mechanism is required to identify which MIP should respond or to an incoming OAM message.

We propose to use the same approach as presented above for MEPs with some variations. It is important to note that "M", merge block of MIP does not prevent OAM packets leaking out as native frames. On edge interfaces, MEPs MUST be configured to prevent the leaking of TRILL OAM packets out of the TRILL Campus.

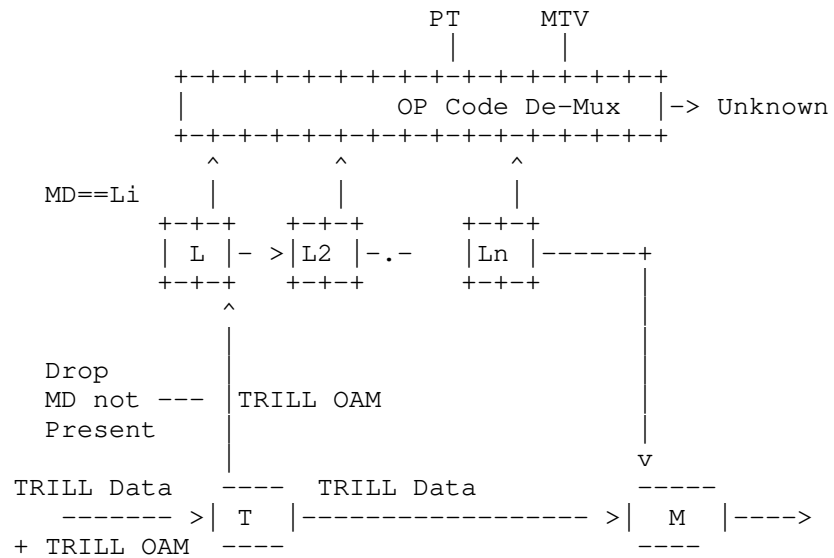


Figure 8 OAM De-Multiplexers at MIP for active SAP

T: TAP processing for MIP. All packets with OAM flag set are captured.

L : MD Level Processing, Packet with matching MD Level are "copied" to the Opcode de-multiplexer and original packet is passed on to the next MD level processor. Other packets are simply passed on to the next MD level processor, without copying to the OP code de-multiplexer.

M : Merge processor, merge OAM packets to be forwarded along with the data flow.

Packets that carry Path Trace (PTM) or Multi-destination Tree Verification (MTV) OpCode are passed on to the respective processors.

Packets with unknown OpCodes are counted and discarded.

7. Approach for Backwards Compatibility

Methodology presented in this document is in-line with the [8021Q] framework or provide fault management coverage. However, in practice, some platforms may not have the required capabilities to support some of the proposed techniques. In this section, we present a method that allows RBridges, which do not have the required hardware capabilities, to participate in the proposed OAM solution.

For backwards compatibility, we propose to locate MEPs and MIPs in the CPU. This will be referred to as the "central brain" model as opposed to "port brain" model.

In the "central brain" model, an RBridge using either ACLs or some other method forwards qualifying OAM messages to the CPU. The CPU then performs the required processing and multiplexing to the correct MP (Maintenance Point).

Additionally, RBridges MUST have the capability to prevent the leaking of OAM packets, as specified in [TRILLOAMREQ] and in the Transmission processing in Figure 9.

Receiver Processing:

```
If (M==1 && F==1) then
    Copy to CPU and Forward normally as defined in RFC 6325
Else if (M==0 && F==1 && egress nickname is the processing RBridge)
then
    Forward to CPU BUT DO NOT forward along the data plane

Else
    Forward as defined in [RFC6325]
End;
```

Transmit Processing:

```
If (F==1) then
    Forward as defined in [RFC6325] BUT Do not de-capsulate and
forward as a native frame
Else
    Forward as defined in [RFC6325]
```

Figure 9 Pseudo code for Backward compatible Processing

[8021Q] requires that the MEP filters or pass through OAM messages based on the MD-Level. The MD-Level is embedded deep in the OAM message. Hence, conventional methods of frame filtering may not be able to filter frames based on the MD-Level. As a result, OAM messages, that must be dropped due to MD level mismatch, may leak in to a TRILL domain with different MD-Level.

This leaking may not cause any functionality loss. Receiving MEP/MIP is required to validate the MD-level prior to acting on the message. Any frames received with an incorrect MD-Level will be dropped.

Generally, TRILL campuses are managed by a single operator, hence there is no risk of security exposure. However, in the event of multi operator deployments, operators should be aware of possible exposure of device specific information and appropriate measures must be taken.

It is also important to note that the MPLS OAM [RFC4379] framework does not include the concept of domains and OAM filtering based on operators. It is our opinion that the lack of OAM frame filtering based on domains does not introduce significant functional deficiency or security risk.

8. Continuity Check Message (CCM)

CCM are used to monitor connectivity and configuration errors. [8021Q] monitors connectivity by listening to periodic CCM messages received from its remote MEP partners in the MA. An [8021Q] MEP identifies cross-connect errors by comparing the MAID in the received CCM message with the MEP's local MAID. The MAID [8021Q] is a 48 byte field that is technology independent. Similarly, the MEPID is a 2 byte field that is independent of the technology. Given this generic definition of CCM fields, CCM as defined in [8021Q] can be utilized in TRILL with no changes. TRILL specific information may be carried in CCMs when encoded using TRILL specific TLVs or sub-TLVs. This is possible since CCMs may carry optional TLVs.

Unlike classical Ethernet environments, TRILL contains multipath forwarding. The path taken by a packet depends on the payload of the packet. The Maintenance Association identifies the interested end-points (MEPs) of a given monitored path. For unicast there are only two MEPs per MA. For multicast there can be two or more MEPs in the MA. Within the MA, we propose to define the entropy values of the monitored flows. CCM transmit logic will utilize these flow entropy values when constructing the CCM packets. Please see section 13. later in the document for the theory of operation of CCM.

We propose to augment the MIB of [8021Q] with definition of flow-entropy. Please see [TRILLOAMMIB] for definition of these and other

Senevirathne Expires August 17, 2013 [Page 22]

TRILL related OAM MIB definitions. Below Figure depicts the correlation between MA, CCM and proposed flow-entropy.

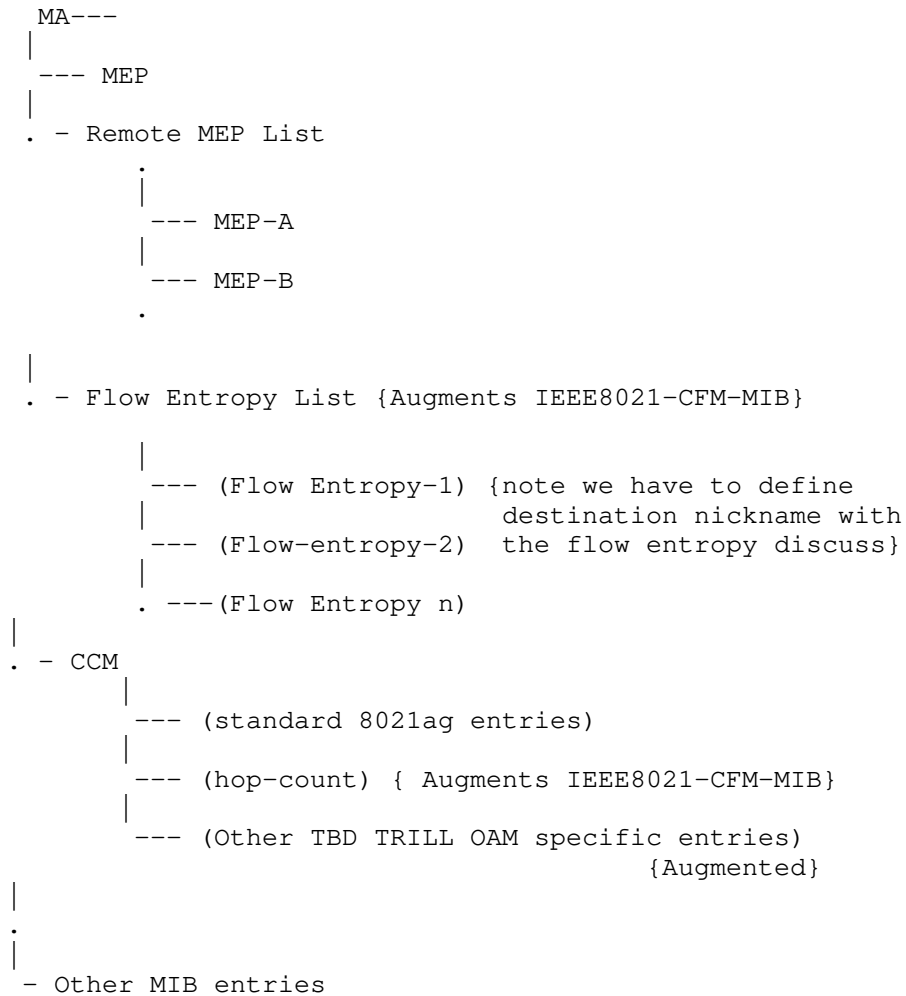


Figure 10 Augmentation of CCM MIB in TRILL

In a multi-pathing environment, a Flow - by definition - is unidirectional. A question may arise as to what flow entropy to be

used in the response. CCMs are unidirectional and have no explicit reply; as such, the issue of the response flow entropy does not arise. In the transmitted CCM, each MEP reports local status using the Remote Defect Indication (RDI) flag. Additionally, a MEP may raise SNMP TRAPS [TRLLOAMMIB] as Alarms when a connectivity failure occurs.

9. TRILL OAM Message Channel

The TRILL OAM Message Channel can be divided into two parts: TRILL OAM Message header and TRILL OAM Message TLVs. Every OAM Message MUST contain a single TRILL OAM message header and a set of one or more specified OAM Message TLVs.

9.1. TRILL OAM Message header

As discussed earlier, we propose to use the message format defined in IEEE 802.1ag. We believe a common messaging framework between [8021Q], TRILL and other similar standards such as Y.1731 can be accomplished by re-using the OAM message header defined in [8021Q].

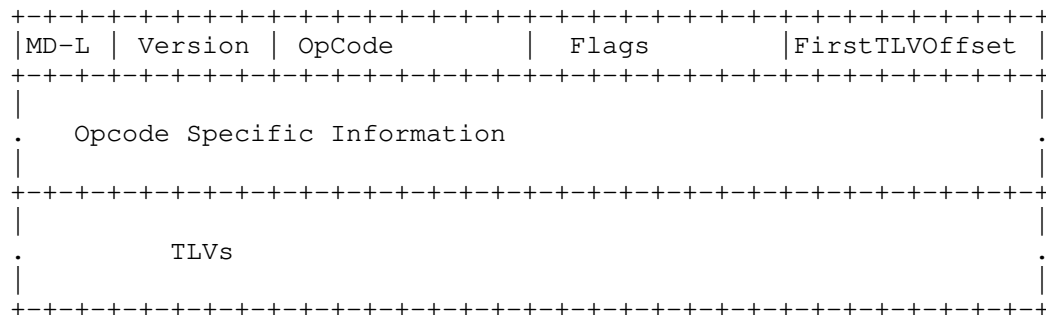


Figure 11 OAM Message Format

- o MD-L: Maintenance Domain Level (3 bits). Identifies the maintenance domain level. For TRILL, this MAY be always set to zero. However, in multilevel TRILL, backbone MAY be of a different MD-LEVEL. (Please refer to [8021Q] for the definition of MD-Level)
- o Version: Indicates the version (5 bits). As specified in [8021Q].

- o **Flags:** Includes operational flags (1 byte). The definition of flags is Opcode-specific and is covered in the applicable sections.
- o **FirstTLVOffset:** Defines the location of the first TLV, in bytes, starting from the end of the FirstTLVOffset field (1 byte). (Refer to [8021Q] for the definition of the FirstTLVOffset.)

MD-L, Version, Opcode, Flags and FirstTLVOffset fields collectively are referred to as the OAM Message Header.

The Opcode specific information section of the OAM Message may contain Session Identification number, time-stamp, etc.

9.2. TRILL OAM Opcodes

The following Opcodes are defined for TRILL. Each of the Opcodes defines a separate TRILL OAM message. Details of the messages are presented in the related sections.

TRILL OAM Message Opcodes:

TBD-64 : Path Trace Reply
 TBD-65 : Path Trace Message
 TBD-66 : Notification Message
 TBD-67 : Multicast Tree Verification Reply
 TBD-68 : Multicast Tree Verification Message

9.3. Format of TRILL OAM TLV

We propose to use the same TLV format as defined in section 21.5.1 of [8021Q]. The following figure depicts the general format of a TRILL OAM TLV:

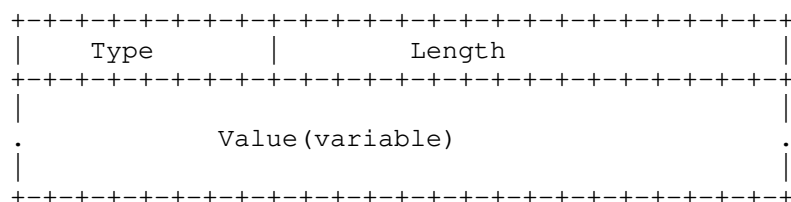


Figure 12 TRILL OAM TLV
 Expires August 17, 2013

Type (1 octet) : Specifies the Type of the TLV (see sections 9.4. for TLV types).

Length (2 octets) : Specifies the length of the 'Value' field in octets. Length of the 'Value' field can be either zero or more octets.

Value (variable): The length and the content of this field depend on the type of the TLV. Please refer to applicable TLV definitions for the details.

Semantics and usage of Type values allocated for TRILL OAM purpose are defined by this document and other future related documents.

9.4. TRILL OAM TLVs

In this section we define TRILL related TLVs. We propose to re-use [8021Q] defined TLVs where applicable. Types 32-63 are reserved for ITU-T Y.1731. We propose to reserve Types 64-95 for TRILL OAM TLVs.

9.4.1. Common TLVs between 802.1ag and TRILL

The following TLVs are defined in [8021Q]. We propose to re-use them where applicable. The format and semantics of the TLVs are as defined in [8021Q]. NOTE: Presented within brackets is the corresponding Type defined in [8021Q].

1. End TLV (0)
2. Sender ID TLV (1)
3. Port Status TLV (2)
4. Data TLV (3)
5. Interface Status TLV (4)
6. Reply Ingress TLV (5)
7. Reply Egress TLV (6)
8. LTM Egress Identifier TLV (7)
9. LTR Egress Identifier TLV (8)
10. Reserved (9-30)
11. Organization specific TLV (31)

9.4.2. TRILL OAM Specific TLVs

As indicated above, Types 64-95 will be requested to be reserved for TRILL OAM purposes. Listed below is a summary of TRILL OAM TLVs and their corresponding codes. Format and semantics of TRILL OAM TLVs are defined in subsequent sections.

1. TRILL OAM Application Identifier (TBD-TLV-64)
2. Out of Band IP Address (TBD_TLV-65)
3. Diagnostic VLAN (TBD-TLV-66)

4. RBridge Scope (TBD-TLV-67)
5. Original Payload (TBD-TLV-68)
6. Previous RBridge Nickname (TBD-TLV-69)
7. TRILL Next Hop RBridge List (ECMP) (TBD-TLV-70)
8. Multicast Receiver Availability (TBD-TLV-71)
9. Flow Identifier (TBD-TLV-72)
10. Reserved (TBD-TLV-72 to TBD-TLV-95)

9.4.2.1. TRILL OAM Application Identifier TLV

TRILL OAM Application Identifier TLV carries TRILL OAM application specific information. The TRILL OAM Application Identifier TLV MUST always be present and MUST be the first TLV in TRILL OAM messages. Messages that do not include the TRILL OAM Application Identifier TLV as the first TLV MUST be discarded by an RBridge, unless that RBridge is running Ethernet CFM.

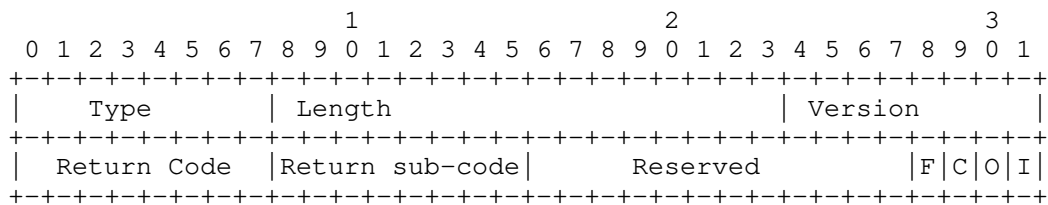


Figure 13 TRILL OAM Message TLV

Type (1 octet) = 64 indicate that this is the TRILL OAM Version

Length (2 octets) = 6

TRILL OAM Version (1 Octet), currently set to zero. Indicates the TRILL OAM version. TRILL OAM version can be different than the [8021Q] version.

Return Code (1 Octet): Set to zero on requests. Set to an appropriate value in response or notification messages.

Return sub-code (1 Octet): Return sub-code is set to zero on transmission of request message. Return sub-code identifies categories within a specific Return code. Return sub-code MUST be interpreted within a Return code.

Reserved: set to zero on transmission and ignored on reception.

F (1 bit) : Final flag, when set, indicates this is the last response.

C (1 bit) : Label error (VLAN/Label mapping error), if set indicates that the label (VLAN/FGL) in the flow entropy is different than the label included in the diagnostic TLV. This field is ignored in request messages and MUST only be interpreted in response messages.

O (1 bit) : If set, indicates, OAM out-of-band response requested.

I (1 bit) : If set, indicates, OAM in-band response requested.

NOTE: When both O and I bits are set to zero, indicates that no response is required (silent mode). User MAY specify both O and I or one of them or none.

9.4.3. Out Of Band Reply Address TLV

Out of Band Reply Address TLV specifies the address to which an out of band OAM reply message MUST be sent. When O bit in the TRILL Version TLV is not set, Out of Band Reply Address TLV is ignored.

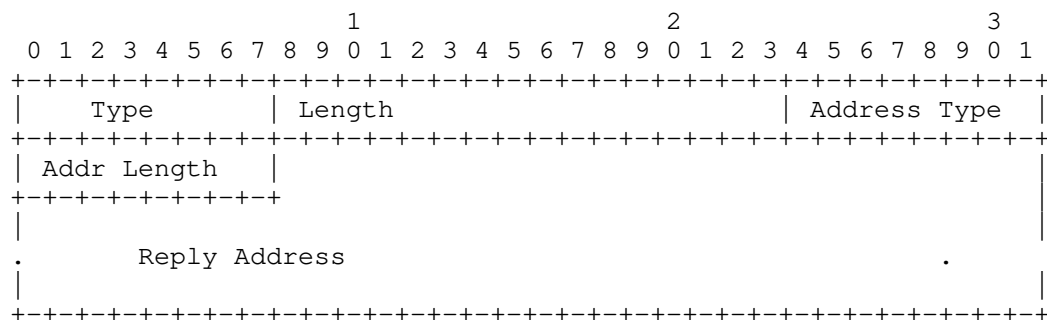


Figure 14 Out of Band IP Address TLV

Type (1 octet) = 64

Length (2 octets) = Variable. Minimum length is 2.

Address Type (1 Octet): 0 - IPv4. 1 - IPv6. 2- TRILL RBridge nickname. All other values reserved.

Addr Length (1 Octet). 4 - IPv4. 16 - IPv6, 2 - TRILL RBRidge nickname.

Reply Address (variable): Address where the reply needed to be sent. Length depends on the address specification.

9.4.3.1. Diagnostics Label TLV

Diagnostic label specifies the data label (VLAN or FGL) in which the OAM messages are generated. Receiving RBridge MUST compare the data label of the Flow entropy to the data label specified in the Diagnostic Label TLV. Label Error Flag in the response (TRILL OAM Message Version TLV) MUST be set when the two VLANs do not match.

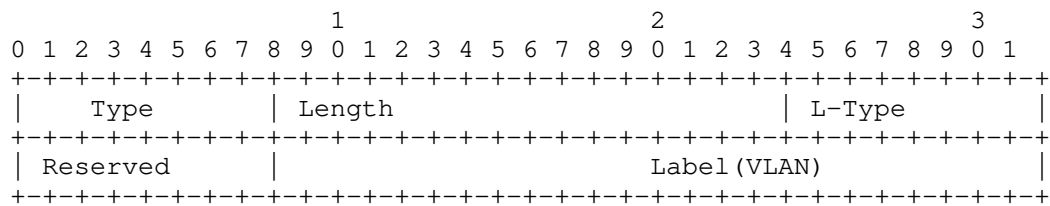


Figure 15 Diagnostic VLAN TLV

Type (1 octet) = 65 indicates that this is the TRILL Diagnostic VLAN TLV

Length (2 octets) = 5

L-Type (Label type, 1 octet)

0- indicate 802.1Q 12 bit VLAN.

1 - indicate TRILL 24 bit fine grain label

Label (24 bits): Either 12 bit VLAN or 24 bit fine grain label.

RBridges do not perform Label error checking when Label TLV is not included in the OAM message. In certain deployment intermediate devices may perform label (VLAN) translation. In such scenarios, originator should not include the diagnostic Label TLV in OAM messages. Inclusion of diagnostic TLV will generated unwanted label error notifications.

9.4.3.2. Original Data Payload TLV

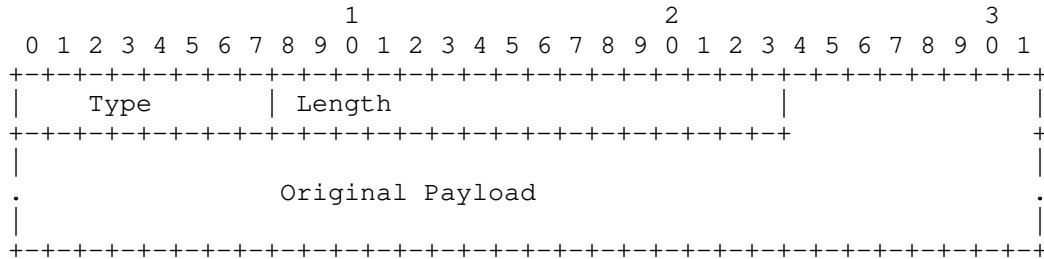


Figure 16 Out of Band IP Address TLV

Length (2 octets) = variable

9.4.3.3. RBridge scope TLV

RBridge scope TLV identifies nicknames of RBridges from which a response is required. RBridge scope TLV is only applicable to Multicast Tree Verification messages. This TLV SHOULD NOT be included in other messages. Receiving RBridges MUST ignore this TLV on messages other than Multicast Verification Message.

Each TLV can contain up to 255 nicknames of in scope RBridges. A Multicast Verification Message may contain multiple "RBridge scope TLVs", in the event that more than 255 in scope RBridges need to be specified.

Absence of the "RBridge scope TLV" indicates that a response is needed from all the RBridges. Please see section 12. for details.

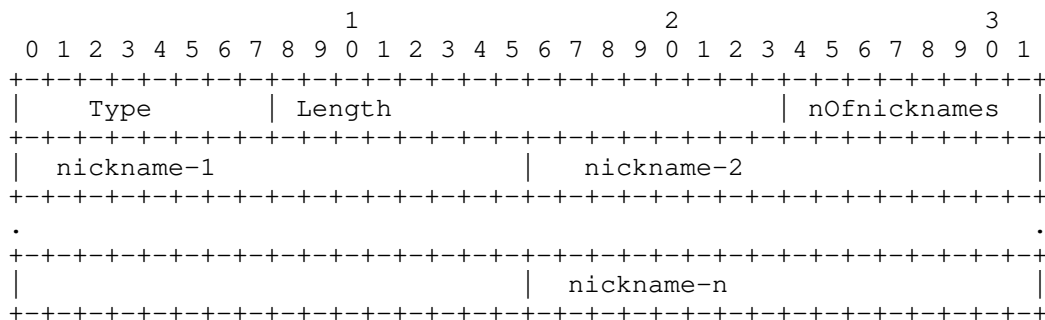


Figure 17 RBridge Scope TLV

Length (2 octets) = variable. Minimum value is 2.

Nickname (2 octets) = 16 bit RBridge nickname.

9.4.3.4. Previous RBridge nickname TLV

"Previous RBridge nickname TLV" identifies the nickname or nicknames of the upstream RBridge. [RFC6325] allows a given RBridge to hold multiple nicknames.

"Upstream RBridge nickname TLV" is an optional TLV. Multiple instances of this TLV MAY be included when an upstream RBridge is represented by more than 255 nicknames (highly unlikely).

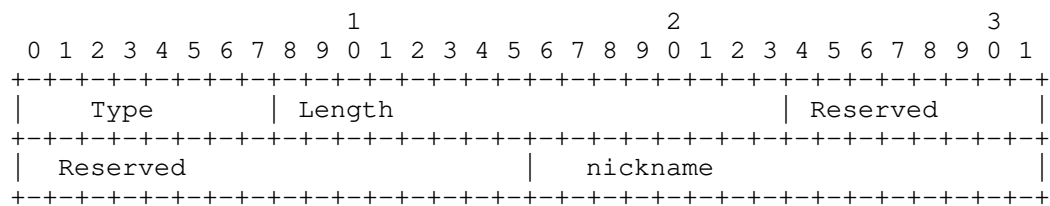


Figure 18 Upstream RBridge nickname TLV

Type (1 octet) = 69 indicates that this is the "Upstream RBridge nickname"

Length (2 octets) = 4.

Nickname (2 octets) = 16 bit RBridge nickname.

9.4.3.5. Next Hop RBridge List TLV

"Next Hop RBridge List TLV" identifies the nickname or nicknames of the downstream next hop RBridges. [RFC6325] allows a given RBridge to have multiple Equal Cost Paths to a specified destination. Each next hop RBridge is represented by one of its nicknames.

"Next Hop RBridge List TLV" is an optional TLV. Multiple instances of this TLV MAY be included when there are more than 255 Equal Cost Paths to the destination.

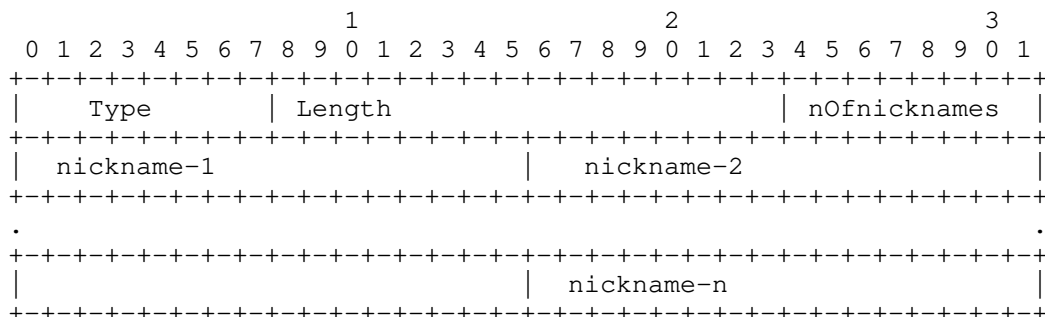


Figure 19 Next Hop RBridge List TLV

Type (1 octet) = 70 indicates that this is the "Next nickname"

Length (2 octets) = variable. Minimum value is 2.

Nickname (2 octets) = 16 bit RBridge nickname.

9.4.3.6. Multicast Receiver Port count TLV

"Multicast Receiver Port Count TLV" identifies the number of ports interested in receiving the specified multicast stream within the responding RBridge on the VLAN specified by the Diagnostic VLAN TLV.

Multicast Receiver Port count is an Optional TLV.

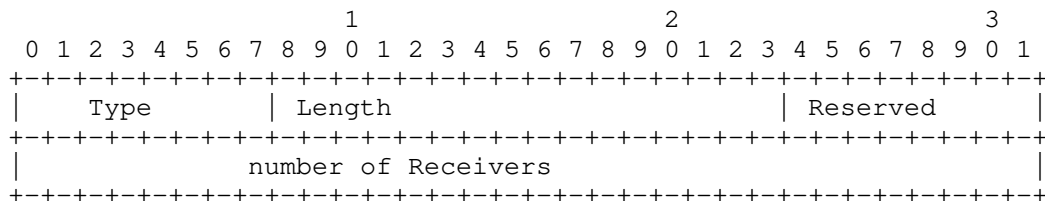


Figure 20 Multicast Receiver Availability TLV

Type (1 octet) = 71 indicates that this is the "Multicast Availability TLV"

Length (2 octets) = 5.

Number of Receivers (4 octets) = Indicates the number of Multicast receivers available on the responding RBridge on the VLAN specified by the diagnostic VLAN.

9.4.4. Flow Identifier (flow-id) TLV

Flow Identifier (flow-id) uniquely identifies a specific flow. The flow-id value is unique per MEP and needed to be interpreted as such.

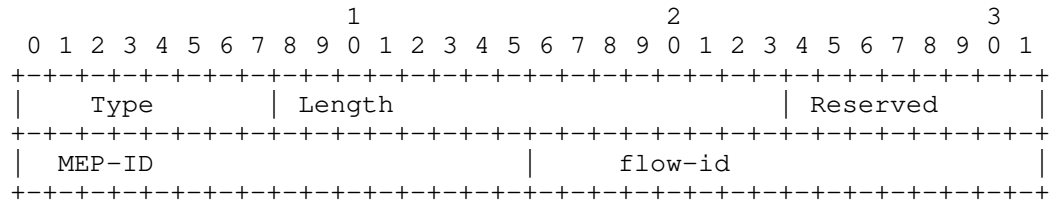


Figure 21 Out of Band IP Address TLV

Type (1 octet) = 72

Length (2 octets) = 5.

Reserved (1 octet) set to 0 on transmission and ignored on reception.

MEP-ID (2 octets) = MEP-ID of the originator [8021Q].

Flow-id (2 octets) = uniquely identifies the flow per MEP. Different MEP may allocate the same flow-id value. The {MEP-ID,flow-id} pair is globally unique.

Inclusion of the MEP-ID in the flow-id TLV allows inclusion of MEP-ID for messages that does not contain MEP-ID in OAM header. Applications may use MEP-ID information for different purposes of troubleshooting.

10. Loopback Message

10.1.1. Loopback OAM Message format

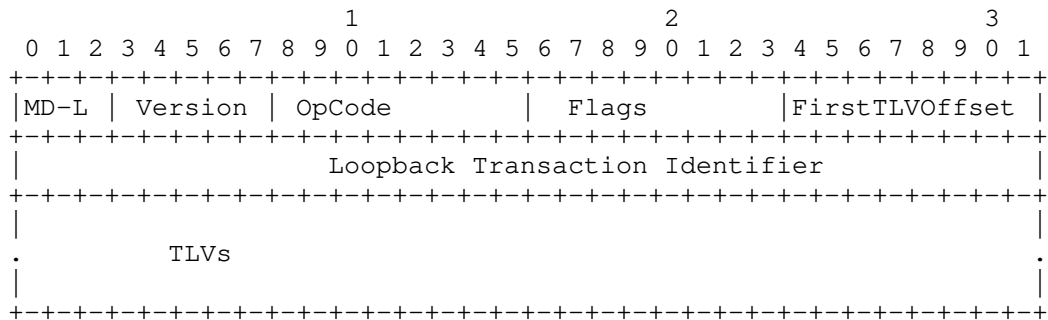


Figure 22 Loopback OAM Message Format

The above figure depicts the format of the Loopback Request and response messages as defined in [8021Q]. The Opcode for Loopback Message is set to 65 and the Opcode for the Reply Message is set to 64. Session Identification Number is a 32-bit integer that allows the requesting RBridge to uniquely identify the corresponding session. Responding RBridges, MUST echo the received "Loopback Transaction Identifier" number without modification.

10.1.2. Theory of Operation

10.1.2.1. Originator RBridge

Originator RBridge Identifies the destination RBridge nickname based on user specification or based on location of the specified destination inner MAC address.

Constructs the flow entropy based on user specified parameters or implementation specific default parameters.

Constructs the TRILL OAM header: Set the opcode to Loopback message type (3). Assign applicable Loopback Transaction Identifier number for the request.

TRILL OAM Version TLV MUST be included and with the flags set to applicable values.

Include following OAM TLVs, where applicable

- o Out-of-band Reply address TLV

- o Diagnostic Label TLV
- o Sender ID TLV

Specify the Hop count of the TRILL data frame per user specification or utilize an applicable Hop count value.

Dispatch the OAM frame for transmission.

RBridge may continue to retransmit the request at periodic intervals, until a response is received or the re-transmission count expires. At each transmission Session Identification number MUST be incremented.

10.1.2.2. Intermediate RBridge

Intermediate RBridges forward the frame as a normal data frame and no special handling is required.

10.1.2.3. Destination RBridge

If the Loopback message is addressed to the local RBridge and satisfies the OAM identification criteria specified in section 3.1. then, the RBridge data plane forwards the message to the CPU for further processing.

TRILL OAM application layer further validates the received OAM frame by examining the presence of OAM-Ethertype at the end of the flow entropy and the MD Level. Frames that do not contain OAM-Ethertype at the end of the flow entropy MUST be discarded.

Construction of the TRILL OAM response:

TRILL OAM application encodes the received TRILL header and flow entropy in the Original payload TLV and includes it in the OAM message.

Set the Return Code and Return sub code to applicable values. Update the TRILL OAM opcode to 2 (Loopback Message Reply)

Optionally, if the VLAN/FGL identifier value of the received flow entropy differs from the value specified in the diagnostic Label, set the Label Error Flag on TRILL OAM Application Identifier TLV.

Include the sender ID TLV (1)

If in-band response was requested, dispatch the frame to the TRILL data plane with request-originator RBridge nickname as the egress RBridge nickname.

If out-of-band response was requested, dispatch the frame to the IP forwarding process.

11. Path Trace Message

The primary use of the Path Trace Message is for fault isolation. It may also be used for plotting the path taken from a given RBridge to another RBridge.

[8021Q] accomplishes the objectives of the TRILL Path Trace Message using Link Trace Messages. Link Trace Messages utilize a well-known multicast MAC address. This works for [8021Q], because for 802.1 both the unicast and multicast paths are congruent. However, TRILL is multicast and unicast incongruent. Hence, we propose TRILL OAM to utilize a new message format: the Path Trace message.

The Path Trace Message has the same format as Loopback Message. Opcode for Path Trace Reply Message is 65 and Request 64

Operation of Path Trace message is identical to Loopback message except, that it is first transmitted with a TRILL Hop count field value of 1. Sending RBridge expects a Time Expiry Return-Code from the next hop or a successful response. If a Time Expiry Return-code is received as the response, the originator RBridge records the information received from intermediate node that generated the Time Expiry message and resends the message by incrementing the previous Hop count value by 1. This process is continued until, a response is received from the destination RBridge or Path Trace process timeout occur or Hop count reaches a configured maximum value.

11.1.1. Theory of Operation

11.1.1.1. Originator RBridge

Identify the destination RBridge based on user specification or based on location of the specified MAC address.

Construct the flow entropy based on user specified parameters or implementation specific default parameters.

Construct the TRILL OAM header: Set the opcode to Path Trace Request message type (65). Assign applicable Session Identification number for the request. Return-code and sub-code MUST be set to zero.

TRILL OAM Application Identifier TLV MUST be included and set the flags to applicable values.

Include following OAM TLVs, where applicable

- o Out-of-band IP address TLV
- o Diagnostic Label TLV
- o Include the Sender ID TLV

Specify the Hop count of the TRILL data frame as 1 for the first request.

Dispatch the OAM frame to the TRILL data plane for transmission.

An RBridge may continue to retransmit the request at periodic intervals, until a response is received or the re-transmission count expires. At each new re-transmission, the Session Identification number MUST be incremented. Additionally, for responses received from intermediate RBridges, the RBridge nickname and interface information MUST be recorded.

11.1.1.2. Intermediate RBridge

Path Trace Messages transit through Intermediate RBridges transparently, unless Hop-count has expired.

TRILL OAM application layer further validates the received OAM frame by examining the presence of TRILL OAM Flag and OAM-Ethertype at the end of the flow entropy and by examining the MD Level. Frames that do not contain OAM-Ethertype at the end of the flow entropy MUST be discarded.

Construction of the TRILL OAM response:

TRILL OAM application encodes the received TRILL header and flow entropy in the Original payload TLV and include it in the OAM message.

Set the Return Code to (2) "Time Expired" and Return sub code to zero (0). Update the TRILL OAM opcode to 64 (Path Trace Message Reply).

If the VLAN/FGL identifier value of the received flow entropy differs from the value specified in the diagnostic Label, set the Label Error Flag on TRILL OAM Application Identifier TLV.

Include following TLVs

Upstream RBridge nickname TLV (69)

Reply Ingress TLV (5)

Reply Egress TLV (6)

Interface Status TLV (4)

TRILL Next Hop RBridge (Repeat for each ECMP) (70)

Sender ID TLV (1)

If Label error detected, set C flag (Label error detected) in the version.

If in-band response was requested, dispatch the frame to the TRILL data plane with request-originator RBridge nickname as the egress RBridge nickname.

If out-of-band response was requested, dispatch the frame to the standard IP forwarding process.

11.1.1.3. Destination RBridge

Processing is identical to section 11.1.1.2. With the exception that TRILL OAM Opcode is set to Path Trace Reply (64).

12. Multi-Destination Tree Verification (MTV) Message

Multi-Destination Tree Verification messages allow verifying TRILL distribution tree integrity and pruning. TRILL VLAN/FGL and multicast pruning are described in [RFC6325] [RFCclcorrect] and [RFCfgl]. Multi-destination tree verification and Multicast group verification messages are designed to detect pruning defects. Additionally, these tools can be used for plotting a given multicast tree within the TRILL campus.

Multi-Destination tree verification OAM frames are copied to the CPU of every intermediate RBridge that is part of the distribution tree being verified. The originator of the Multi-destination Tree verification message, specifies the scope of RBridges from which a response is required. Only, the RBridges listed in the scope field respond to the request. Other RBridges silently discard the request. Inclusion of scope parameter is required to prevent receiving a large number of responses. Typical scenario of distribution tree verification or group verification involves verifying multicast connectivity to selected set of end-nodes as opposed to the entire network. Availability of the scope facilitates narrowing down the focus to only the interested RBridges.

Implementations MAY choose to rate-limit CPU bound multicast traffic. As a result of rate-limiting or due to other congestion conditions, MTV messages may be discarded from time to time by the intermediate RBRidges and the requester may be required to retransmit the request. Implementations SHOULD narrow the embedded scope of retransmission request only to RBRidges that have failed to respond.

12.1. Multi-Destination Tree Verification (MTV) OAM Message Format

Format of MTV OAM Message format is identical to that of Loopback Message format defined in section 10. with the exception that the Loopback Transaction Identifier, in section 10.1.1. , is replaced with the Session Identifier.

12.2. Theory of Operation

12.2.1. Originator RBridge

User is required at minimum to specify either the distribution trees that need to be verified, or Multicast MAC address and VLAN/FGL, or VLAN/FGL and Multicast destination IP address. Alternatively, for more specific multicast flow verification, the user MAY specify more information e.g. source MAC address, VLAN/FGL, Destination and Source IP addresses. Implementations, at a minimum, must allow the user to specify a choice of distribution trees, Destination Multicast MAC address and VLAN/FGL that needed to be verified. Although, it is not mandatory, it is highly desired to provide an option to specify the scope. It should be noted that the source MAC address and some other parameters may not be specified if the Backwards Compatibility Method of section 3.2 is used to identify the OAM frames.

Default parameters MUST be used for unspecified parameters. Flow entropy is constructed based on user specified parameters and/or default parameters.

Based on user specified parameters, the originating RBridge identifies the nickname that represent the multicast tree.

Obtain the applicable Hop count value for the selected multicast tree.

Construct TRILL OAM message header and include Session Identification number. Session Identification number facilitate the originator to map the response to the correct request.

TRILL OAM Application Identifier TLV MUST be included.

Op-Code MUST be specified as Multicast Tree Verification Message (70)

Include RBridge scope TLV (67)

Optionally, include following TLV, where applicable

- o Out-of-band IP address
- o Diagnostic Label
- o Sender ID TLV (1)

Specify the Hop count of the TRILL data frame per user specification. Or utilize the applicable Hop count value, if TRILL Hop count is not being specified by the user.

Dispatch the OAM frame to the TRILL data plane to be ingressed for transmission.

RBridge may continue to retransmit the request at a periodic interval, until a response is received or the re-transmission count expires. At each new re-transmission, the Session Identification number MUST be incremented. At each re-transmission, the RBridge may further reduce the scope to the RBridges that it has not received a response from.

12.2.2. Receiving RBridge

Receiving RBridges identify multicast verification frames per the procedure explained in sections 3.2.

CPU of the RBridge validates the frame and analyzes the scope RBridge list. If the RBridge scope TLV is present and the local RBridge nickname is not specified in the scope list, it will silently discard the frame. If the local RBridge is specified in the scope list OR RBridge scope TLV is absent, the receiving RBridge proceeds with further processing as defined in section 12.2.3.

12.2.3. In scope RBridges

Construction of the TRILL OAM response:

TRILL OAM application encodes the received TRILL header and flow entropy in the Original payload TLV and include in the OAM message.

Set the Return Code to (0) and Return sub code to zero (0). Update the TRILL OAM opcode to 67 (Multicast Tree Verification Reply).

Include following TLVs

Upstream RBridge nickname TLV (69)

Reply Ingress TLV (5)

Interface Status TLV (4)

TRILL Next Hop RBridge (Repeat for each downstream RBridge) (70)

Sender ID TLV (1)

Multicast Receiver Availability TLV (71)

If VLAN cross connect error detected, set C flag (Cross connect error detected) in the version.

If in-band response was requested, dispatch the frame to the TRILL data plane with request-originator RBridge nickname as the egress RBridge nickname.

If out-of-band response was requested, dispatch the frame to the standard IP forwarding process.

13. Application of Continuity Check Message (CCM) in TRILL

Section 8. provides an overview of CCM Messages defined in [8021Q] and how they can be used within the TRILL OAM. In this section, we present the application and Theory of Operations of CCM within the TRILL OAM framework. Readers are referred to [8021Q] for CCM message format and applicable TLV definitions and usages. Only the TRILL specific aspects are explained below.

In TRILL, between any two given MEPs there can be multiple potential paths. Whereas in [8021Q], there is always a single path between any two MEPs, at any given time. [TRILLOAMREQ] requires solutions to have the ability to monitor continuity over one or more paths.

CCM Messages are uni-directional, such that there is no explicit response to a received CCM message. Connectivity status is indicated by setting the applicable flags (e.g. RDI) of the CCM messages transmitted by an MEP.

It is important that the proposed solution accomplishes the requirements specified in [TRILLOAMREQ] within the framework of [8021Q] in a straightforward manner and with minimum changes.

Section 8, above proposed to define multiple flows within the CCM

Senevirathne Expires August 17, 2013 [Page 41]

object, each corresponding to a flow that a given MEP wishes to monitor.

Receiving MEPs do not cross check whether a received CCM belongs to a specific flow from the originating RBridge. Any attempt to track status of individual flows may explode the amount of state information that any given RBridge has to maintain.

Obvious question arises is, how does the originating RBridge knows which flow or flows are at fault?

13.1. CCM Error notification - Method-1

This is accomplished with a combination of RDI flag in the CCM header and SNMP Notifications (Traps).

Each MEP transmits 4 CCM messages per each flow. ([8021Q] detects CCM fault when 3 consecutive CCM messages are lost). Each CCM Message has a unique sequence number.

When an MEP notice a CCM timeout from a remote MEP (MEP-A), it sets the RDI flag on next CCM message it generates. Additionally, it logs and sends SNMP notification that contain the remote MEP Identification, Sequence Number of the last CCM message it received and if available the Sequence Number of the first CCM message it received after the failure. CCM Messages generated by MEP-A has monotonically increasing Sequence Numbers; hence operator can easily identify flows that correspond to specific Sequence Numbers.

Following example illustrate the above.

Assume there are two MEPs, MEP-A and MEP-B.

Assume there are 3 flows between MEP-A and MEP-B.

Lets assume MEP-A allocates sequence numbers as follows

Flow-1 {1,2,3,4,13,14,15,16,.. }

Flow-2 {5,6,7,8,17,18,19,20,.. }

Flow-3 {9,10,12,11,21,22,23,24,.. }

Lets Assume Flow-2 is at fault.

MEP-B, receives CCM from MEP-A with sequence numbers 1,2,3,4, but did not receive 5,6,7,8. CCM timeout is set to 3 CCM intervals in [8021Q]. Hence MEP-B detects the error at 8'th CCM message. At this time the sequence number of the last good CCM message MEP-B has

Senevirathne Expires August 17, 2013 [Page 42]

received from MEP-A is 4. Hence MEP-B will generate an CCM error SNMP notification with MEP-A and Last good sequence number 4.

When MEP-A switch to flow-3 after transmitting flow-2, MEP-B will start receiving CCM messages, in this example it will be CCM message with Sequence Numbers 9,21 and so on. When receipt of a new CCM message from a specific MEP, after a CCM timeout, TRILL OAM will generate SNMP Notification of CCM resume with remote MEP-ID and the first valid Sequence number after the CCM timeout. In the foregoing example, it is MEP-A and Sequence Number 9.

We propose to augment remote MEP list under CCM MIB Object to contain "Last Sequence Number" and "CCM Timeout" variables. Last Sequence Number is updated every time a CCM is received from remote MEP. CCM Timeout variable is set when a CCM timeout has occurred and cleared when a CCM is received. Combination of the two new MIB variables and use of monotonically increasing sequence numbers allow TRILL OAM to clearly identify specific flow or flows at fault.

13.2. CCM Error Notification Method-2

This is accomplished with a combination of RDI flag in the CCM header, flow-id TLV and SNMP Notifications (Traps).

Each MEP transmits 4 CCM messages per each flow. ([8021Q] detects CCM fault when 3 consecutive CCM messages are lost). Each CCM Message has a unique sequence number and unique flow-identifier. The flow identifier is included in the OAM message via flow-id TLV.

When an MEP notice a CCM timeout from a remote MEP (MEP-A), it sets the RDI flag on next CCM message it generates. Additionally, it logs and sends SNMP notification that contain the remote MEP Identification, flow-id and the Sequence Number of the last CCM message it received and if available, the flow-id and the Sequence Number of the first CCM message it received after the failure. Each MEP maintain a unique flow-id per each flow, hence operator can easily identify flows that correspond to the specific flow-id.

Following example illustrate the above.

Assume there are two MEP, MEP-A and MEP-B.

Assume there are 3 flows between MEP-A and MEP-B.

Lets assume MEP-A allocates sequence numbers as follows

Flow-1 Sequence={1,2,3,4,13,14,15,16,.. } flow-id=(1)

Flow-2 Sequence={5,6,7,8,17,18,19,20,.. } flow-id=(2)

Flow-3 Sequence={9,10,12,11,21,22,23,24,... } flow-id=(3)

Lets Assume Flow-2 is at fault.

MEP-B, receives CCM from MEP-A with sequence numbers 1,2,3,4, but did not receive 5,6,7,8. CCM timeout is set to 3 CCM intervals in [8021Q]. Hence MEP-B detects the error at 8'th CCM message. At this time the sequence number of the last good CCM message MEP-B has received from MEP-A is 4 and flow-id of the last good CCM Message is (1). Hence MEP-B will generate a CCM error SNMP notification with MEP-A and Last good flow-id (1) and sequence number 4.

When MEP-A switch to flow-3 after transmitting flow-2, MEP-B will start receiving CCM messages, in this example it will be CCM message with Sequence Numbers 9,10,11,12,21 and so on. When receipt of a new CCM message from a specific MEP, after a CCM timeout, TRILL OAM will generate SNMP Notification of CCM resume with remote MEP-ID and the first valid flow-id and the Sequence number after the CCM timeout. In the foregoing example, it is MEP-A, flow-id (1) and Sequence Number 9.

We propose to augment remote MEP list under CCM MIB Object to contain "Last Sequence Number", flow-id and "CCM Timeout" variables. Last Sequence Number and flow-id are updated every time a CCM is received from a remote MEP. CCM Timeout variable is set when CCM timeout is occurred and cleared when CCM is received.

13.3. Theory of Operation

13.3.1. Originator RBridge

Derive the flow entropy based on flow entropy specified in the CCM Management object.

Construct the TRILL CCM OAM header as specified in [8021Q].

TRILL OAM Version TLV MUST be included as the first TLV and set the flags to applicable values.

Include other TLV specified in [8021Q]

Include following optional TRILL OAM TLVs, where applicable

- o Sender ID TLV

Specify the Hop count of the TRILL data frame per user specification or utilize an applicable Hop count value.

Dispatch the OAM frame to the TRILL data plane for transmission.

RBridge transmits a total of 4 requests, each at CCM retransmission interval. At each transmission Session Identification number MUST be incremented by one.

At the 5 retransmission interval, flow entropy of the CCM packet is updated to the next flow entropy specified in the CCM Management Object. If current flow entropy is the last flow entropy specified, move to the first flow entropy specified and continue the process.

13.3.2. Intermediate RBridge

Intermediate RBridges forward the frame as a normal data frame and no special handling is required.

13.3.3. Destination RBridge

If the CCM Message is addressed to the local RBridge or multicast and satisfies OAM identification methods specified in sections 3.2. then the RBridge data plane forwards the message to the CPU for further processing.

TRILL OAM application layer further validates the received OAM frame by examining the presence of OAM-Ethertype at the end of the flow entropy. Frames that do not contain OAM-Ethertype at the end of the flow entropy MUST be discarded.

Validate the MD-LEVEL and pass the packet to the Opcode de-multiplexer. Opcode de-multiplexer delivers CCM packets to the CCM process.

CCM Process performs processing specified in [8021Q].

Additionally CCM process updates the CCM Management Object with the sequence number of the received CCM packet. Note: Last received CCM sequence number and CCM timeout is tracked per each remote MEP.

If CCM timeout is true for the sending remote MEP, then clear the CCM timeout in the CCM Management object and generate SNMP notification as specified above.

14. Multiple Fragment Reply

Response Message as described in 4.4.2.1 allows Multiple Fragment Reply with use of Final Flag. In case of Multiple Fragment Reply, due to response exceeding MTU size, all messages MUST follow the procedure defined in this section.

All Reply Messages MUST be encoded as described in this document.

Same session Identification Number MUST be included in all related fragments of the same message.

TRILL OAM Application Identifier TLV MUST BE included with the appropriate Final Flag field. Final Flag, MUST, only be set on the final fragment of the reply.

15. Security Considerations

For general TRILL related security considerations, please refer to [RFC6325]. Specific security considerations related methods presented in this document are currently under investigation.

16. Allocation Considerations

16.1. IEEE Allocation Considerations

The IEEE 802.1 Working Group is requested to allocate a separate opcode and TLV space within 802.1QCFM messages for TRILL purpose.

16.2. IANA Considerations

- IANA is requested to allocate a multicast MAC address from the block assigned to TRILL
- Set up sub-registry within the TRILL Parameters registry for block of TRILL OAM OpCodes -
- Set up sub-registry within the TRILL Parameters registry for TRILL OAM TLV Types -
- Set up sub-registry within the TRILL Parameters registry for TRILL OAM return code and return sub codes -
- Request a unicast MAC addressed, reserved for identification of OAM packets discussed in backward compatibility method (section 3.3.)

17. References

17.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.
- [RFC6325] Perlman, R., et.al., "Routing Bridges (R Bridges): Base Protocol Specification", RFC 6325, July 2011.
- [RFCfgl] D. Eastlake, M. Zhang, P. Agarwal, R. Perlman, D. Dutt, "TRILL: Fine-Grained Labeling", draft-ietf-trill-fine-labeling, work in progress.

17.2. Informative References

- [RFC6291] Andersson, L., et.al., "Guidelines for the use of the "OAM" Acronym in the IETF" RFC 6291, June 2011.
- [TRILLOAMMIB] "TRILL OAM MIB", To be published.
- [RFC4379] Kompella, K. et.al, "Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures", RFC 4379, February 2006.
- [TRILLOAMREQ] Senevirathne, T., et.al., "Requirements for Operations, Administration and Maintenance (OAM) in TRILL", draft-ietf-trill-oam-req, Work in Progress, November, 2012.
- [TRILLOAMFM] Salam, S., et.al., "TRILL OAM Framework", draft-ietf-trill-oam-framework, Work in Progress, November, 2012.
- [RFCclcorrect] Eastlake, Donald, et.al. "TRILL: Clarifications, Corrections, and Updates, draft-ietf-trill-clear-correct, July 2012.
- [8021Q] IEEE, "Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks", IEEE Std 802.1Q-2011, August, 2011.

18. Acknowledgments

Work in this document was largely inspired by the directions provided by Stewart Bryant in finding a common OAM solution between SDO.

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

Authors' Addresses

Tissa Senevirathne
CISCO Systems
375 East Tasman Drive.
San Jose, CA 95134
USA.

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

Samer Salam
CISCO Systems
595 Burrard St. Suite 2123
Vancouver, BC V7X 1J1, Canada

Email: ssalam@cisco.com

Deepak Kumar
CISCO Systems
510 McCarthy Blvd,
Milpitas, CA 95035, USA

Phone : +1 408-853-9760
Email: dekumar@cisco.com

Donald Eastlake
Huawei Technologies
155 Beaver Street
Milford, MA 01757

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

Sam Aldrin
Huawei Technologies
2330 Central Express Way
Santa Clara, CA 95951
USA

Email: aldrin.ietf@gmail.com

Yizhou Li
Huawei Technologies
101 Software Avenue,
Nanjing 210012
China

Phone: +86-25-56625375
Email: liyizhou@huawei.com

TRILL Working Group
INTERNET-DRAFT
Intended status: Proposed Standard

Lucy Yong
Donald Eastlake
Sam Aldrin
Huawei Technologies
Jon Hudson
Brocade
February 18, 2013

Expires: August 17, 2013

TRILL Over Pseudo Wires
<draft-yong-pwe3-trill-o-pw-00.txt>

Abstract

This document describes ways to interconnect a pair of TRILL (Transparent Interconnection of Lots of Links) switch ports with two types of pseudo wires under existing TRILL and PWE3 (pseudowire Emulation End-to-End) standards.

Status of This Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

Distribution of this document is unlimited. Comments should be sent to the authors.

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/lid-abstracts.html>. The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

Table of Contents

1. Introduction.....	3
1.1 Conventions used in this document.....	3
2. PWE3 Interconnection of TRILL Switches.....	4
2.1 PWE3 Type Independent Details.....	4
2.2 TRILL over PPP PWE3.....	4
2.3 TRILL over Ethernet PWE3.....	5
2.4 Preferable Pseudowire Type And Auto-Configuration.....	5
3. IANA Considerations.....	6
4. Security Considerations.....	6
Acknowledgements.....	7
Normative References.....	7
Informative References.....	7
Authors' Addresses.....	9

1. Introduction

The IETF has standardized the TRILL (TRansparent Interconnection of Lots of Links) protocol [RFC6325] that provides optimal pair-wise data frame routing without configuration in multi-hop networks with arbitrary topology. TRILL supports multipathing of both unicast and multicast traffic. Devices that implement TRILL are called TRILL Switches or RBridges (Routing Bridges).

End stations are attached to TRILL switches with Ethernet. But links between TRILL switches can be based on arbitrary link protocols, for example PPP [RFC6361], as well as Ethernet [RFC6325]. A set of connected TRILL switches form a TRILL campus which is bounded by end stations and layer 3 routers. Such a campus may contain bridges.

This document specified the use of two types of PWE3 (Pseudowire Emulation End-to-End) pseudowires as links between TRILL switches. It is assumed that such pseudowires are implemented with MPLS.

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].

Acronyms used in this document include the following:

IS-IS - Intermediate System to Intermediate System [IS-IS]
[RFC1195]

MPLS - Multi-Protocol Label Switching

PPP - Point-to-Point Protocol

PW - Pseudowire

PWE3 - PW Emulation End-to-End

RBridge - Routing Bridge, an alternative name for a TRILL Switch

TRILL - Transparent Interconnection of Lots of Links [RFC6325]

TRILL Switch - A device implementing the TRILL protocol

2. PWE3 Interconnection of TRILL Switches

PPP [RFC4618] or Ethernet [RFC4448] pseudowires may be used to interconnect pairs of TRILL switch ports as described below. The pseudowire between such ports can be auto-configured [RFC4447] or manually configured. The TRILL switches, which are TRILL routers, are also acting as label switched routers for those TRILL switch ports.

In both types, the pseudowire provides transparent transport and the two R Bridges appear directly interconnected with a transparent link. With such an interconnection (and negotiation to use TRILL in the PPP case [RFC6361]), the TRILL adjacency over that link is automatically discovered and established through TRILL IS-IS control messages [RFC6325] [RFC6327].

2.1 PWE3 Type Independent Details

The sending pseudowire TRILL switch port MUST copy the priority of the TRILL packets being sent to the 3-bit Class of Service field of the pseudowire label [RFC5462] so the priority will be visible to transit devices that can take the priority into account.

If a pseudowire supports fragmentation and re-assembly, there is no reason to do TRILL MTU testing on it and the pseudowire will not be a constraint on the TRILL campus wide Sz (see Section 4.3.1 [RFC6325]). If the pseudowire does not support fragmentation, then the available TRILL IS-IS packet payload size over the pseudowire (taking into account MPLS encapsulation with a control word) or some lower value, MUST be used in helping to determine Sz (see Section 5 [ClearCorrect]).

An intervening MPLS label switched router or similar device has no awareness of TRILL. Such devices will not change the TRILL Header hop count.

2.2 TRILL over PPP PWE3

For a PPP pseudowire (PW type = 0x0007), the two TRILL switch ports being connected are configured to form a pseudowire with PPP encapsulation [RFC4618]. After the pseudowire is established and TRILL use is negotiated within PPP, the two TRILL switches then appear directly connected with a PPP link [RFC1661].

Behavior for TRILL with a PPP pseudowire continues to follow that of TRILL over PPP as specified in Section 3 of [RFC6361].

2.3 TRILL over Ethernet PWE3

For an Ethernet pseudowire, the two TRILL switch ports being connected are configured to form a pseudowire with Ethernet encapsulation [RFC4448]. The ports MUST use the Raw mode (PW type = 0x0005) and non-service-delimiting, to provide as transparent an Ethernet transport as practical. The two RBridges then appear directly interconnected with an Ethernet link [RFC6325].

Behavior for TRILL with an Ethernet psuedo wire continue to follow that over Ethernet as specified in [RFC6325] and [RFC6327].

2.4 Preferable Pseudowire Type And Auto-Configuration

Use of the PPP pseduowire type is preferable to the Ethernet pseudowire type for the connections discussed in this document. It saves 12 or 16 bytes on every TRILL packet. In particular, the Link Header in the PPP case is simply a 2-byte PPP code point while for the Ethernet case it is 14 or 18 bytes (Outer.MacDA (6), Outer.MacSA (6), sometimes Outer.VLAN (4), and TRILL Ethertype (2)). (While it would also be possible to specify a special custom pseudowire type for TRILL traffic, the authors feel that any efficiency gain over PPP pseudowires would be too small to be worth the complexity of adding such a specification.)

If pseudowire interconnection of two TRILL switch ports is auto-configured [RFC4447] and the initiating RBridge port supports PPP pseudowires, it SHOULD initially attempt the connection set-up with PW type PPP (0x0007). If that pseudowire type is rejected, it SHOULD try again with the Ethernet PW type recommended above (0x0005) if it supports that type. If a responding RBridge port receives a set-up attempt specifying PPP, it SHOULD accept the connection if it supports PPP. If a responding RBridge port receives a set-up attempt specifying Ethernet (PW type = 0x0005), it SHOULD assume that the initiator does not support PPP and accept or reject the Ethernet set-up attempt depending on whether or not it supports Ethernet. SHOULD is specified because local policy as to what pseudowires connections and types are allowed may override these guidelines.

3. IANA Considerations

No IANA action is required by this document. RFC Editor: Please remove this section before publication.

4. Security Considerations

For general TRILL protocol security considerations and those related to Ethernet links, see [RFC6325].

For PPP link TRILL security considerations, see [RFC6361].

For security considerations introduced by carrying Ethernet or PPP TRILL links over pseudowires, see [RFC3985].

Not all implementations need to include specific security mechanisms at the pseudowire layer, for example if they are designed to be deployed only in cases where the networking environment is trusted or where other layers provide adequate security. A complete enumeration of possible deployment scenarios and associated threats and options is not possible and is outside the scope of this document. For applications involving sensitive data, end-to-end security should always be considered, in addition to link security, to provide security in depth.

Acknowledgements

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

Normative References

- [RFC1661] - Simpson, W., Ed., "The Point-to-Point Protocol (PPP)", STD 51, RFC 1661, July 1994.
- [RFC2119] - Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC4447] Martini, L., Ed., Rosen, E., El-Aawar, N., Smith, T., and G. Heron, "Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)", RFC 4447, April 2006.
- [RFC4448] Martini, L., Ed., Rosen, E., El-Aawar, N., and G. Heron, "Encapsulation Methods for Transport of Ethernet over MPLS Networks", RFC 4448, April 2006.
- [RFC4618] Martini, L., "Encapsulation Methods for Transport of PPP/High-Level Data Link Control (HDLC) over MPLS Networks", BCP 116, RFC 4618, September 2006.
- [RFC5462] - Andersson, L. and R. Asati, "Multiprotocol Label Switching (MPLS) Label Stack Entry: "EXP" Field Renamed to "Traffic Class" Field", RFC 5462, February 2009.
- [RFC6325] - Perlman, R., Eastlake 3rd, D., Dutt, D., Gai, S., and A. Ghanwani, "Routing Bridges (Rbridges): Base Protocol Specification", RFC6325, July 2011.
- [RFC6361] - Carlson, J., and D. Eastlake, "PPP Transparent Interconnection of Lots of Links (TRILL) Protocol Control Protocol", RFC6361, August 2011.
- [ClearCorrect] - Eastlake, D., M. Zhang, A. Ghanwani, V. Manral, and A. Banerjee, "TRILL: Clarifications, Corrections, and Updates", draft-ietf-trill-clear-correct, in RFC Editor's queue.

Informative References

- [IS-IS] International Organization for Standardization, "Intermediate system to Intermediate system intra-domain routing information exchange protocol for use in conjunction with the protocol for

providing the connectionless-mode Network Service (ISO 8473)",
ISO/IEC10589:2002, Second Edition, Nov 2002

[RFC1195] - Callon, R., "Use of OSI IS-IS for routing in TCP/IP and
dual environments", RFC 1195, December 1990.

[RFC3985] - Bryant, S., Ed., and P. Pate, Ed., "Pseudo Wire Emulation
Edge-to-Edge (PWE3) Architecture", RFC 3985, March 2005.

[RFC6327] - Eastlake 3rd, D., Perlman, R., Ghanwani, A., Dutt, D.,
and V. Manral, "Routing Bridges (RBridges): Adjacency", RFC
6327, July 2011.

Authors' Addresses

Lucy Yong
Huawei R&D USA
5340 Legacy Drive
Plano, TX 75025 USA

Phone: +1-469-227-5837
Email: lucy.yong@huawei.com

Donald E. Eastlake, 3rd
Huawei R&D USA
155 Beaver Street
Milford, MA 01757 USA

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

Sam Aldrin
Huawei R&D USA
2330 Central Expressway
Santa Clara, CA 95050 USA

Phone: +1-408-330-4517
Email: sam.aldrin@huawei.com

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

Phone: +1-408-333-4062
jon.hudson@brocade.com

Copyright and IPR Provisions

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License. The definitive version of an IETF Document is that published by, or under the auspices of, the IETF. Versions of IETF Documents that are published by third parties, including those that are translated into other languages, should not be considered to be definitive versions of IETF Documents. The definitive version of these Legal Provisions is that published by, or under the auspices of, the IETF. Versions of these Legal Provisions that are published by third parties, including those that are translated into other languages, should not be considered to be definitive versions of these Legal Provisions. For the avoidance of doubt, each Contributor to the IETF Standards Process licenses each Contribution that he or she makes as part of the IETF Standards Process to the IETF Trust pursuant to the provisions of RFC 5378. No language to the contrary, or terms, conditions or rights that differ from or are inconsistent with the rights and licenses granted under RFC 5378, shall have any effect and shall be null and void, whether published or posted by such Contributor, or included with or in such Contribution.

INTERNET-DRAFT
Intended Status: Proposed Standard
Expires: August 22, 2013

Mingui Zhang
Donald Eastlake
Huawei
February 18, 2013

Problem Statement: TRILL Active/Active Edge
draft-zhang-trill-aggregation-03.txt

Abstract

This document specifies TRILL active/active edge which allows multiple Rbridges concurrently forward data frames of the same VLAN on links bundled by Link Aggregation. With this kind of connection, end nodes may increase the bandwidth and reliability of the access at the edge of TRILL campuses. It's required that no loop or duplication is caused by this new connection type. Besides this basic requirement, this document outlines other potential issues associated with TRILL active/active edge and investigates how these issues may be addressed.

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at
<http://www.ietf.org/lid-abstracts.html>

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

Copyright and License Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal

Provisions Relating to IETF Documents
(<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Acronyms and Terminology	3
2.1. Acronyms	4
2.2. Terminology	4
3. Overview	4
4. Frame Processing	6
4.1. Unicast Ingressing	6
4.2. Unicast Egressing	6
4.3. Multicast Ingressing	6
4.4. Multicast Egressing	6
5. DRB and Pseudonode	7
6. MAC Addresses Sharing	8
7. Failures and Self-healing	9
7.1. Link Failure	9
7.2. Node Failure	9
8. Reverse Path Forwarding Check	9
9. Security Considerations	11
10. IANA Considerations	11
11. References	11
11.1. Normative References	11
11.2. Informative References	11
Author's Addresses	12

1. Introduction

TRILL makes use of the ISIS link state routing to provide least cost paths between TRILL switches (a.k.a. Routing Bridge, RBridge). When a multi-access LAN link connects end-stations to multiple RBridges, a single RBridge has to be appointed as the frame forwarder for each VLAN-x on this LAN link. Other RBridges MAY be appointed as frame forwarders for other VLANs but MUST be inhibited from forwarding frames for the same VLAN-x on this LAN link [RFC6349].

A LAG link can also be used to connect end-stations to multiple RBridges. There are two possible scenarios: (a) an end-station is connected to multiple RBridges by a LAG link directly; (b) end-stations are attached to a bridge and this bridge uses a LAG link to connect multiple RBridges. A LAG link may choose any component link to forward frames and never forwards between them. Therefore, it requires the up-connected RBridges to provide active/active attachment instead of the active/standby mode adopted in the Appointed Forwarder mechanism [RFC6349]. This kind of attachment allows end nodes increase the bandwidth and reliability of their access to the TRILL campus via Link Aggregation.

Similar as a LAN link, a LAG link can be represented by a pseudonode. All member RBridges should report their adjacencies to this pseudonode using LSPs. In this way, RBridges attached to the same LAG link forms an active/active edge group. Other RBridges in the campus communicate with this pseudonode using forwarding paths computed according to ISIS link state routing. No additional add-on characteristics are required.

The baseline requirement is that the active/active edge MUST provide frame forwarding without causing loops or duplications to TRILL campus and the end node. In order to work properly, the TRILL active/active edge has to conduct several other issues. The purpose of this document is to outline these issues while specific solutions to address them are to be explored in the future as building blocks of the whole TRILL active/active edge mechanism.

The rest of this document is organized as follows. Section 2 gives acronyms and terminology. Section 3 provides an overview. Section 4 specifies the frame processing behaviors of member RBridges. Section 5 describes how pseudonode is set up. Section 6 explains the MAC sharing among member RBridges. Section 7 describes the self-healing issue. Section 8 investigates how to go through Reverse Path Forwarding Check without packet loss.

2. Acronyms and Terminology

2.1. Acronyms

ISIS: Intermediate System to Intermediate System
TRILL: TRansparent Interconnection of Lots of Links
AF: Appointed Forwarder
LAG: Link Aggregation
DT: Distribution Tree
RPFC: Reverse Path Forwarding Check

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].

In this document, the term "end node" means the end station or bridge connected to the TRILL active/active edge by Link Aggregation.

This document uses "LAG link" to refer the links bundled together by Link Aggregation. The bundled links are referred as "component links" of the "LAG link".

Familiarity with [RFC6325], [RFC6327], and [RFC6349] is assumed in this document. As in [RFC6325], in this document the word "link" means a "bridged LAN", unless otherwise qualified.

3. Overview

If an end node (end station or bridge) uses a LAG link [802.1AX] to connect multiple edge R Bridges, it's expected that all these R Bridges can ingress and egress frames for the end node. In contrast, if multiple R Bridges are connected to a LAN link, only one of them can be appointed as the frame forwarder for each VLAN-x [RFC6349], as illustrated in Figure 2.1 (a). Other R Bridges will be inhibited from ingressing and egressing frames for VLAN-x.

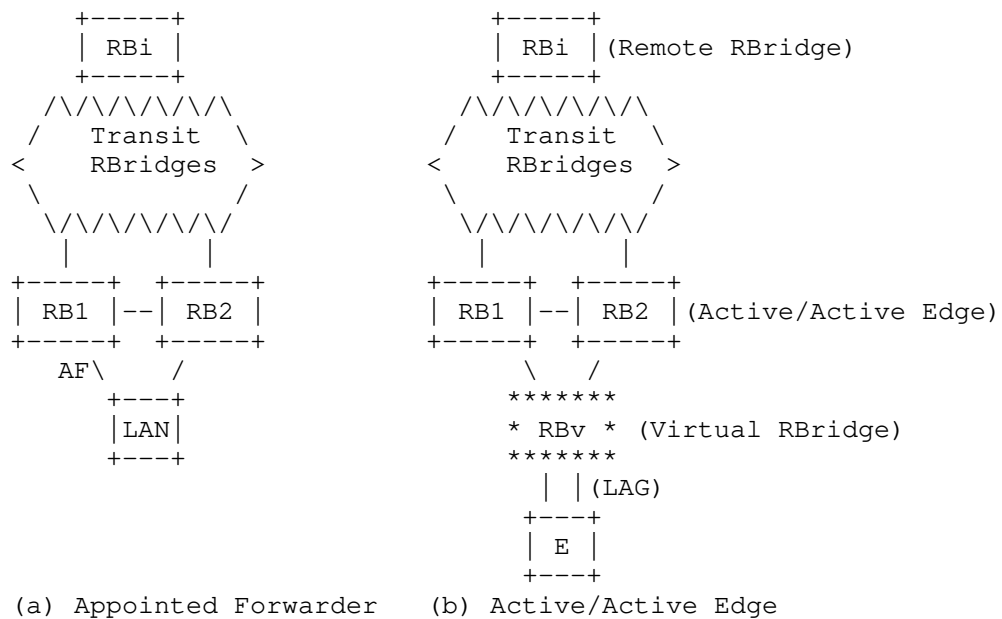


Figure 2.1: TRILL Appointed Forwarder vs Active-Active Edge

As illustrated in Figure 2.1 (b), The end node 'E' are attached to both RB1 and RB2 using a LAG link. Each member RBridge can ingress and egress frames for the end node for VLAN-x. If each of them uses its own nickname as the ingress nickname, the remote RBridge may observe different locations for one MAC address at different time, which is referred as the "MAC move" problem in this document. The MAC move problem affects the path selection at the remote RBridge. Frames destined to the end node may go through different paths, which may cause frame disorder of a traffic flow.

In order to avoid the MAC move problem, each member RBridge should use a uniform nickname as the ingress nickname in TRILL data frame encapsulation. As shown in Figure 2.1 (b), member RBridges pretend there is an virtual RBridge connected to them, acting as the appointed forwarder of the end node. It is naturally to denote this virtual RBridge as a pseudonode. All RBridges connected to the LAG link forms adjacencies with the pseudonode. Other RBridges believe there is an RBridge RBv connecting RB1, RB2. Note that member RBridges SHOULD NOT announce they are VLAN-x Appointed Forwarder if VLAN-x is enabled on the LAG link.

Although the above example includes two edge RBridges, the TRILL active/active edge solution SHOULD support cases with more than two member RBridges.

4. Frame Processing

When the end node injects frames into the TRILL campus via a member RBridge, this RBridge encapsulates the native frames on behalf of the pseudonode. When frames are sent to the end node, the pseudonode is supposed to be the egress RBridge. It's REQUIRED that RBridges other than the active/active members are not aware of the active/active group and need not change their frame processing behavior.

Compared to the Appointed Forwarder mechanism, all active/active member RBridges are able to ingress and egress frames of VLAN-x on the same link. It is crucial to avoid loops and duplications in the frame processing.

4.1. Unicast Ingressing

Receiver RBridges encapsulate native frames using the nickname of the pseudonode as the ingress nickname. When these TRILL data frames arrive at the remote RBridge, the MAC addresses will be learnt from packet decapsulation. The remote RBridge will regard the pseudonode as the egress RBridge for these MAC addresses.

4.2. Unicast Egressing

As learnt in the MAC table, TRILL data frames from remote RBridges destined to the end node will be sent to the pseudonode rather than member RBridges. If member RBridges receive TRILL data frames whose egress RBridge is the pseudonode, they can judge that these frames should be egressed onto the LAG link.

However, member RBridges MUST NOT egress any TRILL data frames whose ingress RBridge is the pseudonode. Otherwise, loops will happen.

4.3. Multicast Ingressing

The end node chooses one component link of the LAG link to send multicast frames to member RBridges. Similar as the unicast ingressing, the receiver RBridge encapsulate the native frames using the nickname of the pseudonode as the ingress nickname.

Different member RBridges MUST NOT share the same Distribution Tree to ingress a multicast frame of a specific VLAN-x from the end node. Otherwise, some multicast frames may suffer from loss due to Reverse Path Forwarding Check. This issues is detailed in Section 8.

4.4. Multicast Egressing

Multicast frames sent along the VLAN-x Distribution Tree may reach

all member RBridges. However, only one of them can egress the multicast frames onto the LAG link. Otherwise, the end node will suffer from frame duplication. This requirement can be met if member RBridges calculate the Distribution Tree regarding the pseudonode as a normal RBridge. Then only one parent RBridge will be selected for the pseudonode. Other non-parent member RBridges MUST refrain from egressing multicast frames of VLAN-x onto the LAG link.

Similar as the unicast egressing, member RBridges MUST NOT egress any multicast frames whose ingress RBridge is the pseudonode.

5. DRB and Pseudonode

As we know, a DRB MAY give a pseudonode name to a LAN link, issue an LSP (Link State PDU) on behalf of the pseudonode, and issues CSNPs (Complete Sequence Number PDUs) on the LAN link [RFC6325]. Different from a LAN link, there is no HELLO exchanging on the LAG link. Thus, the DRB cannot be elected using HELLO protocol. Member RBridges MAY establish a dedicated RBridge Channel to discover each other and elect the DRB (DRB for active/active RBridge group, aDRB) to execute the above tasks: to assign the nickname and issue LSP and CSNPs. The member RBridge with the highest priority to be the tree root is a good choice.

Member RBridges SHOULD be able to discover each other to resolve misconfiguration and failures. Each member RBridge SHALL report their connection to the LAG. The MAC address of the end node MAY be used to identify the LAG to which the member RBridges are connected.

One RBridge may be connected to multiple LAG links. It's probably that all these LAG links share the same set of member RBridges. However, these LAG links MUST NOT share the same pseudonode, otherwise it can cause the following issue.

- o Component Links from Different LAG Links Cannot be Distinguished:
Assume member RBridge RBi is connected to multiple end nodes and these links are all advertised as a single ISIS link "Rbi-RBv". Remote RBridges cannot distinguish these links connecting RBi and RBv. When one of these links fails, it becomes problematic. On one hand, if the failed link is not advertised as a down ISIS link, traffic sent from remote RBridges to RBv via the failed link will be trapped by blackholing. On the other hand, if the failed link is announced as a down ISIS link. Component links from other LAG links will be disconnected mistakenly.

The right choice is to represent every LAG link as a unique pseudonode. In this way, the failure of a component link of a LAG link can be interpreted as an ISIS link failure. Thus the aDRB can

issue a new LSP on half of the pseudonode to trigger the link state update across the campus.

6. MAC Addresses Sharing

When a member RBridge learns a MAC address from the encapsulation or decapsulation of a TRILL data frame, it SHOULD share this learning among all member RBridges. Afterwards, a frame destined to this MAC address can be delivered to the LAG link or ingressed to the TRILL campus by any other member RBridge as a unicast native frame or TRILL data frame.

- a) Northbound Sharing: When a remote RBridge chooses the path to send data frames to the end node, these frames may arrive at anyone of the member RBridges, given that member RBridges may be on the Equal Cost Multiple Paths from the remote RBridge to the pseudonode. If the MAC address from the end node was learnt and recorded by any member RBridge before. The receiver RBridge SHOULD have recorded this MAC (VLAN ID, MAC Address, Port Number) as well, so that the frame can be delivered as a known unicast to the end node. Therefore, local MAC addresses learnt from data frames sent by the end node (northbound) SHOULD be shared among member RBridges.
- b) Southbound Sharing: The end node may choose any component link to inject a frame, which achieves load-balance on the LAG link. If the destination MAC address has been learnt by any member RBridge, the receiver RBridge SHOULD also hold that MAC record (VLAN ID, MAC Address, Egress RBridge Nickname). Thus the data frame need not be sent as a multicast frame (unknown unicast). Therefore, MAC addresses learnt from data frames sent by remote RBridges to the end node (southbound) should be shared as well.

When an RBridge learns a source MAC address from a data frame, it will record the VLAN ID, the source MAC address and location which can be the incoming port number or the ingress nickname. A MAC address shared by a peer RBridge is recorded as if it is locally learned. For example, when RB1 shares a MAC with RB2, RB2 should set the incoming port as its port attaching to the end node.

It is REQUIRED that all member RBridges set the same aging time for each MAC address. Every time a MAC address is learnt or updated, all member RBridges MUST update the record and reset its aging time. It's probably that data frames from one source MAC are received continuously. There is no problem to update the entry of this MAC locally. However, when this update is executed among multiple member RBridges, the intensive updates may consume a considerable bandwidth. Therefore, member RBridges need a communication channel to realize

the MAC sharing, which can be realized through the extension of ESADI or using a dedicated RBridge Channel [Channel].

7. Failures and Self-healing

Resilience is a major purpose that the active/active edge aims to achieve. From the side of the end node, the LAG link provides reliability of the access link. From the side of the member RBridges, the state change of the active/active edge caused by link or node failures is reflected by the update of LSPs of member RBridges. This provides self-healing of the active/active edge.

7.1. Link Failure

The failure of a component link of the LAG link is translated into an ISIS link failure: if a member RBridge is disconnected from the end node, it will send out an LSP to announce that it is not connected to the pseudonode. This will trigger the update of forwarding tables of remote RBridges. Since other member RBridges have also reported the connection to the pseudonode, remote RBridges in the TRILL campus can send frames to the pseudonode via any other member RBridge. Therefore, the reach-ability to the end node is not broken by this link failure.

If the link connecting the aDRB and the end node fails, the link failure will trigger the election of aDRB. The new aDRB SHOULD reuse the nickname allocated to the pseudonode, which avoids changing the locations of MAC addresses from the end node learnt by remote RBridges.

The extreme case is that the last component link of the LAG fails. Then the aDRB SHOULD update its LSPs to remove the pseudonode from the campus, which also destroys the whole active/active edge.

7.2. Node Failure

The node failure of member RBridges will also be reflected by LSP announcement. If the aDRB fails, a new aDRB will be elected and this new aDRB SHOULD reuse the nickname of the pseudonode allocated by the old aDRB.

8. Reverse Path Forwarding Check

Reverse Path Forwarding Check (RPFC) is used by TRILL to suppress forwarding loops of multicast frames [RFC6325]. For a specific Distribution Tree (DT), a multicast frame from a specific ingress RBridge can arrive at only one expected link of an RBridge. RBridges MUST drop multicast frames that fail the RPFC [RFC6325].

When multiple member RBridges ingress multicast frames for VLAN-x of the end node simultaneously, it can not guarantee that these frames always arrive at the expected link of at a remote RBridge. The following example explains this issue.

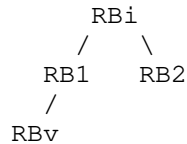
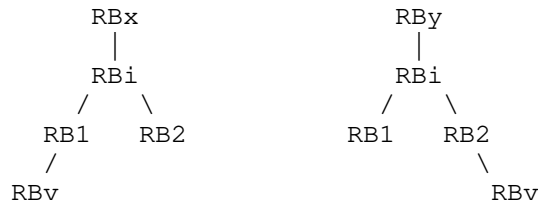


Figure 7.1: The Distribution Tree, root=RBi

Suppose a Distribution Tree of Figure 2.1 (b) is constructed as shown in Figure 7.1. For this Distributions Tree, multicast frames from RBv to RBi is expected to be received at the port attaching to RB1. With the active/active connection, RB2 can receive native data frames from the LAG link as well. If RB2 adopts the above Distribution Tree, multicast frames from RBv to RBi will be received at the port attaching to RB2. This brings the problem: these frames will be discarded according to the rule of RPFC.



(a) DT, root=RBx (b) DT, root=RBv

Figure 7.2: Assign an Unique Tree to each Member RBridge

One way to avoid the above issue is to leverage the feature that RBridges can compute multiple Distribution Trees. Be sure to assign an unique Distribution Tree to each member RBridge for multicast frame distribution. Identify these trees using their root RBridge nicknames. The example in Figure 7.2 illustrates this method, where RB1 and RB2 adopt two different Distribution Trees.

Active/active edge need to assign at least one Distribution Tree per component link of a LAG link, the maximally allowed number of component links depends on the number of Distribution Trees that all RBridges can compute. However, LAGs of the best current practice have two component links, which are well supported by TRILL switches.

In [CMT], the Affinity TLV is used to achieve the above assignment of

Distribution Trees to member RBridges. It is REQUIRED that all RBridges in the campus are able to recognize the Affinity TLV and compute Distribution Trees as this TLV specified.

When there is a link or node failure in the active/active edge, the failed Distribution Tree should be re-allocated to a new member RBridge. It is RECOMMENDED that this re-allocation is incremental. In other words, other Distribution Trees not affected by the failure SHOULD be retained.

9. Security Considerations

This document raises no new security issues for ISIS.

10. IANA Considerations

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

11. References

11.1. Normative References

- [RFC6325] R. Perlman, D. Eastlake, et al, "RBridges: Base Protocol Specification", RFC 6325, July 2011.
- [RFC6349] R. Perlman, D. Eastlake, et al, "RBridges: Appointed Forwarders", RFC 6349, November 2011.
- [Channel] D. Eastlake, V Manral, et al, "TRILL: RBridge Channel Support", draft-ietf-trill-rbridge-channel-08.txt, July 2012, working in progress.
- [CMT] T. Senevirathne, J. Pathangi, et al, "Coordinated Multicast Trees (CMT) for TRILL", draft-ietf-trill-cmt-01.txt, November 2012, working in progress.

11.2. Informative References

- [802.1AX] "IEEE Standard for Local and metropolitan area networks - Link Aggregation", IEEE Std 802.1 AX-2008, 3 November 2008.

Author's Addresses

Mingui Zhang
Huawei Technologies

Email: zhangmingui@huawei.com

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

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