

TRILL WG  
Internet-Draft  
Intended status: Standards Track  
Expires: September 12, 2018

Radia Perlman  
Dell EMC  
Fangwei Hu  
ZTE Corporation  
Donald Eastlake  
Ting Liao  
Huawei Technologies  
Mar 11, 2018

TRILL Smart Endnodes  
draft-ietf-trill-smart-endnodes-11.txt

Abstract

This draft addresses the problem of the size and freshness of the endnode learning table in edge 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 edge R Bridge can distinguish a "Smart Endnode" from a "normal endnode". The Smart Endnode uses the nickname of the attached edge R Bridge, so this solution does not consume extra nicknames. The solution also enables Fine Grained Label aware endnodes.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

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

This Internet-Draft will expire on September 12, 2018.

Copyright Notice

Copyright (c) 2018 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

(<https://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 . . . . .	2
2. Conventions used in this document . . . . .	3
2.1. Terminology . . . . .	3
2.2. Requirements Language . . . . .	4
3. Solution Overview . . . . .	4
4. Smart-Hello Mechanism between Smart Endnode and RBridge . . . . .	5
4.1. Smart-Hello Encapsulation . . . . .	6
4.2. Edge RBridge's Smart-Hello . . . . .	7
4.3. Smart Endnode's Smart-Hello . . . . .	7
5. Data Packet Processing . . . . .	9
5.1. Data Packet Processing for Smart Endnode . . . . .	9
5.2. Data Packet Processing for Edge RBridge . . . . .	10
6. Multi-homing Scenario . . . . .	11
7. Security Considerations . . . . .	12
8. IANA Considerations . . . . .	13
9. Acknowledgements . . . . .	13
10. References . . . . .	13
10.1. Informative References . . . . .	13
10.2. Normative References . . . . .	14
Authors' Addresses . . . . .	15

## 1. Introduction

The IETF TRILL (Transparent Interconnection of Lots of Links) protocol [RFC6325] [RFC7780] 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 [IS-IS] [RFC7176] 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" or "edge TRILL Switch", whereas one that exclusively forwards encapsulated frames is known as a "transit RBridge" or "transit TRILL Switch". An edge RBridge traditionally is the one that encapsulates a native Ethernet frame with a TRILL header, or that receives a TRILL-encapsulated packet and decapsulates the TRILL header. To

encapsulate efficiently, the edge RBridge must keep an "endnode table" consisting of (MAC, Data Label, TRILL egress switch nickname) sets, for those remote MAC addresses in Data Labels currently communicating with endnodes to which the edge RBridge is attached.

These table entries might be configured, received from ESADI [RFC7357], looked up in a directory [RFC7067], or learned from decapsulating received traffic. If the edge RBridge has attached endnodes communicating with many remote endnodes, this table could become very large. Also, if one of the MAC addresses and Data Labels in the table has moved to a different remote TRILL switch, it might be difficult for the edge RBridge to notice this quickly, and because the edge RBridge is encapsulating to the incorrect egress RBridge, the traffic will get lost.

## 2. Conventions used in this document

### 2.1. Terminology

Edge RBridge: An RBridge providing endnode service on at least one of its ports. It is also called an edge TRILL Switch.

Data Label: VLAN or FGL.

DRB: Designated RBridge [RFC6325].

ESADI: End Station Address Distribution Information [RFC7357].

FGL: Fine Grained Label [RFC7172].

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

PDU: Protocol Data Unit.

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

Smart Endnode: An endnode that has the capability specified in this document including learning and maintaining (MAC, Data Label, Nickname) entries and encapsulating/decapsulating TRILL frame.

Transit RBridge: An RBridge exclusively forwards encapsulated frames. It is also called a transit TRILL Switch.

TRILL: Transparent Interconnection of Lots of Links [RFC6325][RFC7780].

TRILL ES-IS: TRILL End System to Intermediate System, is a variation of TRILL IS-IS designed to operate on a TRILL link among and between one or more TRILL switches and end stations on that link[RFC8171].

TRILL Switch: a device that implements the TRILL protocol; an alternative term for an RBridge.

## 2.2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

## 3. Solution Overview

The Smart Endnode solution defined in this document addresses the problem of the size and freshness of the endnode learning table in edge RBridges. An endnode E, attached to an edge RBridge R, tells R that E would like to be a "Smart Endnode", which means that E will encapsulate and decapsulate the TRILL frame, using R's nickname. Because E uses R's nickname, this solution does not consume extra nicknames.

Take Figure 1 as the example Smart Endnode scenario: RB1, RB2 and RB3 are the RBridges in the TRILL domain, and SE1 and SE2 are the Smart Endnodes which can encapsulate and decapsulate the TRILL packets. RB1 is the edge RB that SE1 and SE2 have attached to. RB1 assigns one of its nicknames to be used by SE1 and SE2.

Each Smart Endnode, SE1 and SE2, uses RB1's nickname when encapsulating, and maintains an endnode table of (MAC, label, TRILL egress switch nickname) for remote endnodes that it (SE1 or SE2) is corresponding with. RB1 does not decapsulate packets destined for SE1 or SE2, and does not learn (MAC, label, TRILL egress switch nickname) for endnodes corresponding with SE1 or SE2, but RB1 does decapsulate, and does learn (MAC, label, TRILL egress switch nickname) for any endnodes attached to RB1 that have not declared themselves to be Smart Endnodes.

Just as an RBridge learns and times out (MAC, label, TRILL egress switch nickname), Smart Endnodes SE1 and SE2 also learn and time out endnode entries. However, SE1 and SE2 might also determine, through ICMP messages or other techniques that an endnode entry is not successfully reaching the destination endnode, and can be deleted, even if the entry has not timed out.

If SE1 wishes to correspond with destination MAC D, and no endnode entry exists, SE1 will encapsulate the packet as an unknown destination, or consulting a directory [RFC7067] (just as an RBridge would do if there was no endnode entry).

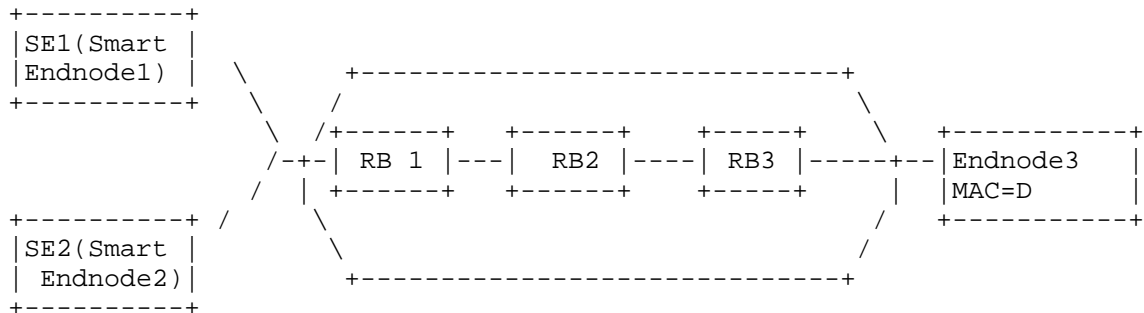


Figure 1 Smart Endnode Scenario

The mechanism in this draft is that the Smart Endnode SE1 issues a Smart-Hello, indicating SE1's desire to act as a Smart Endnode, together with the set of MAC addresses and Data Labels that SE1 owns. The Smart-Hello is used to announce the Smart Endnode capability and parameters (such as MAC address, Data Label etc.). The Smart-Hello is a type of TRILL ES-IS PDU, which is specified in section 5 of [RFC8171]. The detailed content for a Smart Endnode's Smart-Hello is defined in section 4.

If RB1 supports having a Smart Endnode neighbor it also sends Smart-Hellos. The Smart Endnode learns from RB1's Smart-Hellos what RB1's nickname is and which trees RB1 can use when RB1 ingresses multi-destination frames. Although Smart Endnode SE1 transmits Smart-Hellos, it does not transmit or receive LSPs or E-L1FS FS-LSPs [RFC7780].

Since a Smart Endnode can encapsulate TRILL Data packets, it can cause the Inner.Label to be a Fine Grained Label [RFC7172], thus this method supports FGL aware endnodes. When and how a Smart Endnode decides to use the FGL instead of VLANs to encapsulate the TRILL Data packet is out of scope in this document.

#### 4. Smart-Hello Mechanism between Smart Endnode and RBridge

The subsections below describe Smart-Hello messages.

#### 4.1. Smart-Hello Encapsulation

Although a Smart Endnode is not an RBridge, does not send LSPs or maintain a copy of the link state database, and does not perform routing calculations, it is required to have a "Hello" mechanism (1) to announce to edge RBridges that it is a Smart Endnode and (2) to tell them what MAC addresses it is handling in what Data Labels. Similarly, an edge RBridge that supports Smart Endnodes needs a message (1) to announce that support, (2) to inform Smart Endnodes what nickname to use for ingress and what nickname(s) can be used as egress nickname in a multi-destination TRILL Data packet, and (3) the list of Smart Endnodes it knows about on that link.

The messages sent by Smart Endnodes and by edge RBridges that support Smart Endnodes are called "Smart-Hellos". The Smart-Hello is a type of TRILL ES-IS PDU, which is specified in [RFC8171].

The Smart-Hello Payload, both for Smart-Hellos sent by Smart Endnodes and for Smart-Hellos sent by Edge RBridges, consists of TRILL IS-IS TLVs as described in the following two sub-sections. The non-extended format is used so TLVs, sub-TLVs, and APPsub-TLVs have an 8-bit size and type field. Both types of Smart-Hellos MUST include a Smart-Parameters APPsub-TLV as follows inside a TRILL GENINFO TLV:

```

+-----+
|Smart-Parameters|                (1 byte)
+-----+
|   Length   |                (1 byte)
+-----+-----+-----+-----+
|           Holding Time           | (2 bytes)
+-----+-----+-----+-----+
|                   Flags                   | (2 bytes)
+-----+-----+-----+-----+

```

Figure 2 Smart Parameters APPsub-TLV

- o Type: APPsub-TLV type Smart-Parameters, value is TBD1.
- o Length: 4.
- o Holding Time: A time in seconds as an unsigned integer. It has the same meaning as the Holding Time field in IS-IS Hellos [IS-IS]. A Smart Endnode and an Edge RBridge supporting Smart Endnodes MUST send a Smart-Hello at least three times during their Holding Time. If no Smart-Hellos is received from a Smart Endnode or Edge RBridge within the most recent Holding Time it sent, it is assumed that it is no longer available.

o Flags: At this time all of the Flags are reserved and MUST be send as zero and ignored on receipt.

If more than one Smart Parameters APPsub-TLV appears in a Smart-Hello, the first one is used and any following ones are ignored. If no Smart Parameters APPsub-TLV appears in a Smart-Hello, that Smart-Hello is ignored.

#### 4.2. Edge RBridge's Smart-Hello

The edge RBridge's Smart-Hello contains the following information in addition to the Smart-Parameters APPsub-TLV:

o RBridge's nickname. The nickname sub-TLV, specified in section 2.3.2 in [RFC7176], is reused here carried inside a TLV 242 (IS-IS router capability) in a Smart-Hello frame. If more than one nickname appears in the Smart-Hello, the first one is used and the following ones are ignored.

o Trees that RBl can use when ingressing multi-destination frames. The Tree Identifiers Sub-TLV, specified in section 2.3.4 in [RFC7176], is reused here.

o Smart Endnode neighbor list. The TRILL Neighbor TLV, specified in section 2.5 in [RFC7176], is reused for this purpose.

An Authentication TLV MAY also be included.

#### 4.3. Smart Endnode's Smart-Hello

A new APPsub-TLV (Smart-MAC TLV) is defined for use by Smart Endnodes as defined below. In addition, there will be a Smart-Parameters APPsub-TLV and there MAY be an Authentication TLV in a Smart Endnode Smart-Hello.

If there are several VLANs/FGL Data Labels for that Smart Endnode, the Smart-MAC APPsub-TLV is included several times in Smart Endnode's Smart-Hello. This APPsub-TLV appears inside a TRILL GENINFO TLV.

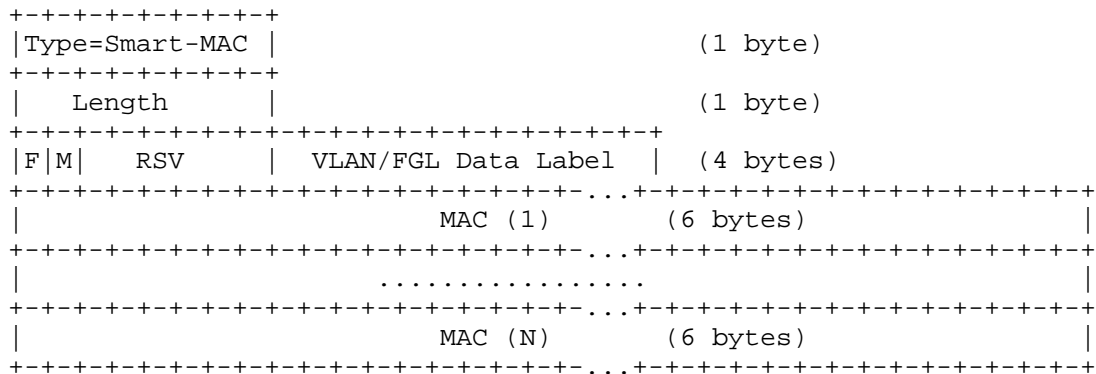


Figure 3 Smart-MAC APPsub-TLV

- o Type: TRILL APPsub-TLV Type Smart-MAC, value is TBD2.
- o Length: Total number of bytes contained in the value field of the TLV, that is, the sum of the length of the F/M/RSV/FGL Data Label fields and 6 times the number of MAC addresses present. So, if there are n MAC addresses, this is  $4+6*n$ .
- o F: 1 bit. If it is set to 1, it indicates that the endnode supports FGL data labels [RFC7172], and that this Smart-MAC APPsub-TLV has an FGL in the following VLAN/FGL field. Otherwise, the VLAN/FGL Data Label field is a VLAN ID. (See below for the format of the VLAN/FGL Data Label field).
- o M: 1 bit. If it is set to 1, it indicates multi-homing (See Section 6). If it is set to 0, it indicates that the Smart Endnodes are not using multi-homing.
- o RSV: 6 bits, is reserved for the future use.
- o VLAN/FGL Data Label: 24bits. If F is 1, this field is a 24-bit FGL Data Label for all subsequent MAC addresses in this APPsub-TLV. Otherwise, if F is 0, the lower 12 bits is the VLAN of all subsequent MAC addresses in this APPsub-TLV, and the upper 12 bits is not used (sent as zero and ignored on receipt). If there is no VLAN/FGL data label specified, the VLAN/FGL Data Label is zero.
- o MAC(i): This is a 48-bit MAC address reachable in the Data Label sent by the Smart Endnode that is announcing this APPsub-TLV.



## 5. Data Packet Processing

The subsections below specify Smart Endnode data packet processing. All TRILL Data packets sent to or from Smart Endnodes are sent in the Designated VLAN [RFC6325] of the local link but do not necessarily have to be VLAN tagged.

### 5.1. Data Packet Processing for Smart Endnode

A Smart Endnode does not issue or receive LSPs or E-L1FS FS-LSPs or calculate topology. It does the following:

- o A Smart Endnode maintains an endnode table of (the MAC address of remote endnode, Data Label, the nickname of the edge RBridge's attached) entries of end nodes with which the Smart Endnode is communicating. Entries in this table are populated the same way that an edge RBridge populates the entries in its table:
  - \* learning from (source MAC address ingress nickname) on packets it decapsulates.
  - \* by querying a directory [RFC7067].
  - \* by having some entries configured.
- o When Smart Endnode SE1 wishes to send unicast frame to remote node D, if (MAC address of remote endnode D, Data Label, nickname) entry is in SE1's endnode table, SE1 encapsulates the ingress nickname as the nickname of the RBridge(RB1), egress nickname as indicated in D's table entry. If D is unknown, SE1 either queries a directory or encapsulates the packet as a multi-destination frame, using one of the trees that RB1 has specified in RB1's Smart-Hello. The mechanism for querying a directory is given in [RFC8171].
- o When SE1 wishes to send a BUM packet to the TRILL campus, SE1 encapsulates the packet using one of the trees that RB1 has specified.

If the Smart Endnode SE1 sends a multi-destination TRILL Data packet, the destination MAC of the outer Ethernet is the All-RBridges multicast address.

The Smart Endnode SE1 need not send Smart-Hellos as frequently as normal RBridges. These Smart-Hellos could be periodically unicast to the Appointed Forwarder RB1. In case RB1 crashes and restarts, or the DRB changes and SE1 receives the Smart-Hello without mentioning SE1, SE1 SHOULD send a Smart-Hello immediately. If RB1 is Appointed

Forwarder for any of the VLANs that SE1 claims, RB1 MUST list SE1 in its Smart-Hellos as a Smart Endnode neighbor.

## 5.2. Data Packet Processing for Edge RBridge

The attached edge RBridge processes and forwards TRILL Data packets based on the endnode property rather than for encapsulation and forwarding the native frames the same way as the traditional RBridges. There are several situations for the edge RBridges as follows:

- o If receiving an encapsulated unicast TRILL Data packet from a port with a Smart Endnode, with RB1's nickname as ingress, the edge RBridge RB1 forwards the frame to the specified egress nickname, as with any encapsulated frame. However, RB1 SHOULD filter the encapsulation frame based on the inner source MAC and Data Label as specified for the Smart Endnode. If the MAC (or Data Label) are not among the expected entries of the Smart Endnode, the frame would be dropped by the edge RBridge. If the edge RBridge does not perform this check, it makes it easier for a rogue end station to inject bogus TRILL Data packets into the TRILL campus.
- o If receiving a unicast TRILL Data packet with RB1's nickname as egress from the TRILL campus, and the destination MAC address in the enclosed packet is a MAC address that has been listed by a "Smart Endnode", RB1 leaves the packet encapsulated to that Smart Endnode. The outer Ethernet destination MAC is the destination Smart Endnode's MAC address, the inner destination MAC address is either the Smart Endnode's MAC address or some other MAC address that the Smart Endnode advertised in its Smart Hello, and the outer Ethernet source MAC address is the RB1's port MAC address. The edge RBridge still decreases the Hop count value by 1, for there is one hop between the RB1 and Smart Endnode.
- o If receiving a multi-destination TRILL Data packet from a port with a Smart Endnode, RBridge RB1 forwards the TRILL encapsulation to the TRILL campus based on the distribution tree indicated by the egress nickname. If the egress nickname does not correspond to a distribution tree, the packet is discarded. If there are any normal endnodes (i.e, non-Smart Endnodes) attached to the edge RBridge RB1, RB1 decapsulates the frame and sends the native frame to these ports possibly pruned based on multicast listeners, in addition to forwarding the multi-destination TRILL frame to the rest of the campus.
- o If RB1 receives a native multi-destination data frame, which is sent by a non-Smart Endnode, from a port, including hybrid endnodes (Smart Endnodes and non-Smart Endnodes), RB1 will encapsulate it as

multi-destination TRILL Data packet , and send the encapsulated multi-destination TRILL Data Packet out that same port to the Smart Endnodes attached to the port, and also send the encapsulated multi-destination TRILL Data Packet to the TRILL campus through other ports.

o If RB1 receives a multi-destination TRILL Data packet from a remote RBridge, and the exit port includes hybrid endnodes(Smart Endnodes and non-Smart Endnodes), it sends two copies of multicast frames out the port, one as native and the other as TRILL encapsulated frame. When Smart Endnode receives multi-destination TRILL Data packet, it learns the remote (MAC address, Data Label, Nickname) entry. A Smart Endnodes ignores native data frames. A normal (non-Smart) Endnode receives the native frame and learns the remote MAC address and ignores the TRILL data packet. This transit solution may bring some complexity for the edge RBridge and waste network bandwidth resource, so avoiding the hybrid endnodes scenario by attaching the Smart Endnodes and non-Smart Endnodes to different ports is RECOMMENDED.

## 6. Multi-homing Scenario

Multi-homing is a common scenario for the Smart Endnode. The Smart Endnode is on a link attached to the TRILL domain in two places: to edge RBridge RB1 and RB2. Take the figure below as example. The Smart Endnode SE1 is attached to the TRILL domain by RB1 and RB2 separately. Both RB1 and RB2 could announce their nicknames to SE1.

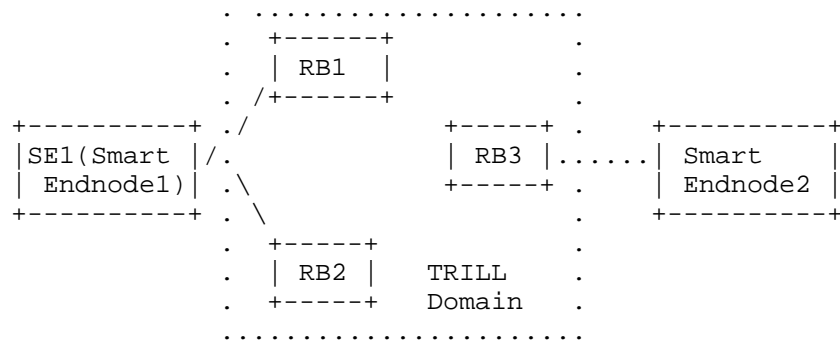


Figure 4 Multi-homing Scenario

Smart Endnode SE1 can choose either RB1 or RB2's nickname, when encapsulating and forwarding a TRILL data packet. If the active-active load balance is considered for the multi-homing scenario, the Smart Endnode SE1 could use both RB1 and RB2's nickname to encapsulate and forward TRILL Data packet. SE1 uses RB1's nickname

when forwarding through RB1, and RB2's nickname when forwarding through RB2. This will cause MAC flip-flopping(see [RFC7379]) of the endnode table entry in the remote RBridges (or Smart Endnodes). The solution for the MAC flip-flopping issue is to set a multi-homing bit in the RSV field of the TRILL data packet. When remote RBridge RB3 or Smart Endnodes receives a data packet with the multi-homed bit set, the endnode entries (SE1's MAC address, label, RB1's nickname) and (SE1's MAC address, label, RB2's nickname) will coexist as endnode entries in the remote RBridge. (An alternative solution would be to use the ESADI protocol to distribute multiple attachments of a MAC address of a multi-homing group, The ESADI is deployed among the edge RBridges (See section 5.3 of [RFC7357])).

## 7. Security Considerations

Smart-Hellos can be secured by using Authentication TLVs based on [RFC5310]. If they are not secured, then it is easier for a rogue end station that does not possess the required keying material to be falsely recognized as a valid Smart Endnode.

For general TRILL Security Considerations, see [RFC6325]. As stated there, since end stations are connected to edge RBridge ports by Ethernet, those ports MAY require end stations to authenticate themselves using [IEEE802.1X] and authenticate and encrypt traffic to/from the RBridge port with [IEEE802.1AE].

If they misbehave, Smart Endnodes can forge arbitrary ingress and egress nicknames in the TRILL Headers of the TRILL Data packets they construct. Decapsulating at egress RBridges or remote Smart Endnodes that believe such a forged ingress nickname would send future traffic destined for the inner source MAC address of the TRILL Data frame to the wrong edge RBridge if data plane learning is in use. Because of this, an RBridge port should not be configured to support Smart Endnodes unless the end stations on that link are trusted or can be adequately authenticated.

As with any end station, Smart Endnodes can forge the outer MAC addresses of packets they send (See Section 6 of [RFC6325].) Because they encapsulate TRILL Data packets, they can also forge inner MAC addresses. The encapsulation performed by Smart Endnodes also means they can send data in any Data Label which means they must be trusted in order to enforce a security policy based on Data Labels.

The TRILL-Hello is a type of TRILL ES-IS, and is defined in [RFC8171]. Receiving and processing TRILL-Hello for RBridges and Smart Endnodes would not bring more security and vulnerability issues than the TRILL ES-IS security defined in [RFC8171].

For added security against the compromise of data due to its mis-delivery for any reason, including the above, end-to-end encryption and authentication should be considered; that is, encryption and authentication from source end station to destination end station.

The mechanism described in this document requires Smart Endnodes to be aware of the MAC address(es) of the TRILL edge RBridge(s) to which they are attached and the egress RBridge nickname from which the destination of the packets is reachable. With that information, Smart Endnodes can learn a substantial amount about the topology of the TRILL domain. Therefore, there could be a potential security risk when the Smart Endnodes are not trusted or are compromised.

## 8. IANA Considerations

IANA is requested to allocate APPsub-TLV type numbers for the Smart-MAC and Smart-Parameters APPsub-TLVs from the range below 256 and update the "TRILL APPsub-TLV Types under IS-IS TLV 251 Application Identifier 1" registry as follows.

Protocol	Description	Reference
TBD1	Smart-Parameters	[this document]
TBD2	Smart-MAC	[this document]

Table 1

## 9. Acknowledgements

The contributions of the following persons are gratefully acknowledged: Mingui Zhang, Weiguo Hao, Linda Dunbar, Kesava Vijaya Krupakaran and Andrew Qu.

## 10. References

### 10.1. Informative References

- [IEEE802.1AE]  
"IEEE Standard for Local and metropolitan area networks--Media Access Control (MAC) Security.", 2006.
- [IEEE802.1X]  
"IEEE Standard for Local and metropolitan area networks--Port-Based Network Access Control", 2010.

- [RFC7067] Dunbar, L., Eastlake 3rd, D., Perlman, R., and I. Gashinsky, "Directory Assistance Problem and High-Level Design Proposal", RFC 7067, DOI 10.17487/RFC7067, November 2013, <<https://www.rfc-editor.org/info/rfc7067>>.
- [RFC7379] Li, Y., Hao, W., Perlman, R., Hudson, J., and H. Zhai, "Problem Statement and Goals for Active-Active Connection at the Transparent Interconnection of Lots of Links (TRILL) Edge", RFC 7379, DOI 10.17487/RFC7379, October 2014, <<https://www.rfc-editor.org/info/rfc7379>>.

## 10.2. Normative References

- [IS-IS] ISO/IEC 10589:2002, Second Edition,, "Intermediate System to Intermediate System Intra-Domain Routing Exchange Protocol for use in Conjunction with the Protocol for Providing the Connectionless-mode Network Service (ISO 8473)", 2002.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC5310] Bhatia, M., Manral, V., Li, T., Atkinson, R., White, R., and M. Fanto, "IS-IS Generic Cryptographic Authentication", RFC 5310, DOI 10.17487/RFC5310, February 2009, <<https://www.rfc-editor.org/info/rfc5310>>.
- [RFC6325] Perlman, R., Eastlake 3rd, D., Dutt, D., Gai, S., and A. Ghanwani, "Routing Bridges (RBridges): Base Protocol Specification", RFC 6325, DOI 10.17487/RFC6325, July 2011, <<https://www.rfc-editor.org/info/rfc6325>>.
- [RFC7172] Eastlake 3rd, D., Zhang, M., Agarwal, P., Perlman, R., and D. Dutt, "Transparent Interconnection of Lots of Links (TRILL): Fine-Grained Labeling", RFC 7172, DOI 10.17487/RFC7172, May 2014, <<https://www.rfc-editor.org/info/rfc7172>>.
- [RFC7176] Eastlake 3rd, D., Senevirathne, T., Ghanwani, A., Dutt, D., and A. Banerjee, "Transparent Interconnection of Lots of Links (TRILL) Use of IS-IS", RFC 7176, DOI 10.17487/RFC7176, May 2014, <<https://www.rfc-editor.org/info/rfc7176>>.

- [RFC7357] Zhai, H., Hu, F., Perlman, R., Eastlake 3rd, D., and O. Stokes, "Transparent Interconnection of Lots of Links (TRILL): End Station Address Distribution Information (ESADI) Protocol", RFC 7357, DOI 10.17487/RFC7357, September 2014, <<https://www.rfc-editor.org/info/rfc7357>>.
- [RFC7780] Eastlake 3rd, D., Zhang, M., Perlman, R., Banerjee, A., Ghanwani, A., and S. Gupta, "Transparent Interconnection of Lots of Links (TRILL): Clarifications, Corrections, and Updates", RFC 7780, DOI 10.17487/RFC7780, February 2016, <<https://www.rfc-editor.org/info/rfc7780>>.
- [RFC8171] Eastlake 3rd, D., Dunbar, L., Perlman, R., and Y. Li, "Transparent Interconnection of Lots of Links (TRILL): Edge Directory Assistance Mechanisms", RFC 8171, DOI 10.17487/RFC8171, June 2017, <<https://www.rfc-editor.org/info/rfc8171>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

## Authors' Addresses

Radia Perlman  
Dell EMC  
176 South Street  
Hopkinton, MA 01748  
USA

Phone: +1-206-291-367  
Email: [radiaperlman@gmail.com](mailto:radiaperlman@gmail.com)

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

Phone: +86 21 68896273  
Email: [hu.fangwei@zte.com.cn](mailto:hu.fangwei@zte.com.cn)

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

Phone: +1-508-634-2066  
Email: d3e3e3@gmail.com

Ting Liao  
Huawei Technologies  
Nanjing, Jiangsu 210012  
China

Email: liaoting1@huawei.com