Internet Engineering Task Force (IETF)
Request for Comments: 8383
Category: Standards Track
ISSN: 2070-1721
May 2018

W. Hao
D. Eastlake, 3rd
Y. Li
Huawei
M. Umair
Cisco

Transparent Interconnection of Lots of Links (TRILL): Address Flush Message

Abstract

The TRILL (Transparent Interconnection of Lots of Links) protocol, by default, learns end station addresses from observing the data plane. In particular, it learns local Media Access Control (MAC) addresses and the edge switch port of attachment from the receipt of local data frames and learns remote MAC addresses and the edge switch port of attachment from the decapsulation of remotely sourced TRILL Data packets.

This document specifies a message by which a TRILL switch can explicitly request other TRILL switches to flush certain MAC reachability learned through the decapsulation of TRILL Data packets. This is a supplement to the TRILL automatic address forgetting (see Section 4.8.3 of RFC 6325) and can assist in achieving more rapid convergence in case of topology or configuration change.

Status of This Memo

Table of Contents

1. Introduction 3
1.1. Terminology and Abbreviations 3
2. Address Flush Message Details 5
2.1. VLAN Block Only Case 6
2.2. Extensible Case 8
2.2.1. Blocks of VLANs 12
2.2.2. Bit Map of VLANs 12
2.2.3. Blocks of FGLs 13
2.2.4. list of FGLs 13
2.2.5. Big Map of FGLs 14
2.2.6. All Data Labels 14
2.2.7. MAC Address List 15
2.2.8. MAC Address Blocks 16
3. IANA Considerations 17
3.1. Address Flush RBridge Channel Protocol Number 17
3.2. TRILL Address Flush TLV Types 17
4. Security Considerations 17
5. References 18
5.1. Normative References 18
5.2. Informative References 19
Acknowledgements 19
Authors' Addresses 20

1. Introduction

By default, edge TRILL (Transparent Interconnection of Lots of Links) switches [RFC6325] [RFC7780], also called edge Routing Bridges (RBridges), learn end station MAC address reachability from observing the data plane. On receipt of a native frame from an end station, they would learn the local MAC address attachment of the source end station. And on egressing (decapsulating) a remotely originated TRILL Data packet, they learn the remote MAC address and remote attachment TRILL switch. Such learning is all scoped by data label (VLAN or Fine-Grained Label (FGL) [RFC7172]).

TRILL has mechanisms for timing out such learning and appropriately clearing it based on some network connectivity and configuration changes; however, there are circumstances under which it would be helpful for a TRILL switch to be able to explicitly flush (purge) certain learned end station reachability information in remote RBridges to achieve more-rapid convergence. Section 6.2 of [RFC4762] is an example of the use of such a mechanism.

Another example, based on Appendix A.3 of [RFC6325] ("Wiring Closet Topology"), presents a bridged LAN connected to a TRILL network via multiple RBridge ports. For optimum paths, Appendix A.3.3 suggests configuring the RBridge ports to be like one Spanning Tree Protocol (STP) tree root in the bridged LAN. The Address Flush message in this document could also be triggered in this case when one of the edge RBridges receives Topology Change (TC) information (e.g., TC in STP, Topology Change Notification (TCN) in Multiple Spanning Tree Protocol (MSTP)) in order to rapidly flush the MAC addresses for specific VLANs learned at the other edge RBridge ports.

A TRILL switch can easily flush any locally learned addresses it wants. This document specifies an RBridge Channel Support protocol [RFC7178] message to request flushing address information for specific VLANs or FGLs ([RFC7172]) learned from decapsulating TRILL Data packets.

1.1. Terminology and Abbreviations

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

This document uses the terms and abbreviations defined in [RFC6325] and [RFC7178] as well as the following:

Data Label: A VLAN or FGL

Edge TRILL Switch: A TRILL switch attached to one or more links that provide end station service

FCS: Frame Check Sequence

FGL: Fine-Grained Label [RFC7172]

Management VLAN: A VLAN in which all TRILL switches in a campus indicate interest so that multi-destination TRILL Data packets, including RBridge Channel protocol messages [RFC7178], sent with that VLAN as the Inner.VLAN will be delivered to all TRILL switches in the campus. Usually, no end station service is offered in the Management VLAN.

MAC: Media Access Control

RBridge: An alternative name for a TRILL switch

STP: Spanning Tree Protocol

TC: Topology Change message

TCN: Topology Change Notification message

TRILL switch: A device implementing the TRILL protocol [RFC6325] [RFC7780]

2. Address Flush Message Details

The Address Flush message is an RBridge Channel protocol message [RFC7178].

The general structure of an RBridge Channel packet on a link between TRILL switches is shown in Figure 1. The Protocol field in the RBridge Channel Header gives the type of RBridge Channel packet and indicates how to interpret the Channel-Protocol-Specific Payload [RFC7178].

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

Figure 1: RBridge Channel Protocol Message Structure

By default, an Address Flush RBridge Channel protocol message applies to addresses within the Data Label that appear right after the Inner Ethernet Addresses. Address Flush protocol messages are usually sent as multi-destination packets (TRILL Header M bit equal to one) so as to reach all TRILL switches offering end station service in the VLAN or FGL specified by that Data Label. Both multi-destination and unicast Address Flush messages SHOULD be sent at priority 6 since they are important control messages but are lower priority than control messages that establish or maintain adjacency.

Nevertheless:

- There are provisions for optionally indicating the Data Label(s) to be flushed for cases where the Address Flush message is sent over a Management VLAN or the like.

- An Address Flush message can be sent unicast, if it is desired to clear addresses at one TRILL switch only.

- An Address Flush message can be sent selectively to the RBridges that have at least one access port configured as one of the VLANs or FGLs specified in the Address Flush message payload.

Implementations should consider logging Address Flush messages received with appropriate protections against packet storms.

2.1. VLAN Block Only Case

Figure 2 expands the RBridge Channel Header and Channel-Protocol-Specific Payload from Figure 1 for the case of the VLAN-only-based Address Flush message. This form of the Address Flush message is optimized for flushing MAC addresses based on nickname and blocks of VLANs. 0x8946 is the Ethertype assigned by IEEE for the RBridge Channel protocol [RFC7178].

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
RBridge Channel Header:
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |    RBridge-Channel (0x8946)   |  0x0  |Channel Protocol= 0x009|
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |          Flags        |  ERR  |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
Address Flush Protocol Specific:
   +-+-+-+-+-+-+-+-+
   | K-nicks       |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | Nickname 1                    | Nickname 2                    |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | Nickname ...                  | Nickname K-nicks              |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | K-VLBs        |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | RESV  | Start.VLAN 1          | RESV  | End.VLAN 1            |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | RESV  | Start.VLAN 2          | RESV  | End.VLAN 2            |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | RESV  | Start.VLAN ...        | RESV  | End.VLAN ...          |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | RESV  | Start.VLAN K-VLBs     | RESV  | End.VLAN K-VLBs       |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: Address Flush Message - VLAN Block Case

The fields in Figure 2 related to the Address Flush message are as follows:

Channel Protocol: The RBridge Channel Protocol value allocated for Address Flush (see Section 3).

K-nicks: The number of nicknames listed as an unsigned integer. If this is zero, the ingress nickname in the TRILL Header [RFC6325] is considered to be the only nickname to which the message applies. If non-zero, it gives the number of nicknames listed right after K-nicks to which the message applies, and, in this non-zero case, the flush does not apply to the ingress nickname in the TRILL Header unless it is also listed. The message flushes address learning due to egressing TRILL Data packets that had an ingress nickname to which the message applies.

Nickname: A listed nickname to which it is intended that the Address Flush message apply. If an unknown or reserved nickname occurs in the list, it is ignored, but the address flush operation is still executed with the other nicknames. If an incorrect nickname occurs in the list, so that some address learning is flushed that should not have been flushed, the network will still operate correctly; however, it will be less efficient as the incorrectly flushed learning is relearned.

K-VLBs: The number of VLAN blocks present as an unsigned integer. If this byte is zero, the message is the more general format specified in Section 2.2. If it is non-zero, it gives the number of blocks of VLANs present. Thus, in the VLAN Block address flush case, K-VLBs will be at least one.

RESV: 4 reserved bits. MUST be sent as zero and ignored on receipt.

Start.VLAN, End.VLAN: These 12-bit fields give the beginning and ending VLAN IDs of a block of VLANs. The block includes both the starting and ending values; so, a block of size one is indicated by setting End.VLAN equal to Start.VLAN. If Start.VLAN is 0x000, it is treated as if it was 0x001. If End.VLAN is 0xFFF, it is treated as if it was 0xFFE. If End.VLAN is smaller than Start.VLAN, considering both as unsigned integers, that VLAN block is ignored, but the address flush operation is still executed with other VLAN blocks in the message. VLAN blocks may overlap, in which case, the address flush operation is applicable to a VLAN covered by any one or more of the blocks in the message.

This message flushes all addresses in an applicable VLAN learned from egressing TRILL Data packets with an applicable nickname as ingress. To flush addresses for all VLANs, it is easy to specify a block covering all valid VLAN IDs (i.e., from 0x001 to 0xFFE).

2.2. Extensible Case

A more general form of the Address Flush message is provided to support flushing by FGL and more efficient encodings of VLANs and FGLs where using a set of contiguous blocks is cumbersome. It also supports optionally specifying the MAC addresses to clear. This form is extensible.

The extensible case is indicated by a zero in the byte shown in Figure 2 as "K-VLBs" followed by other information encoded as TLVs.

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
RBridge Channel Header:
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |    RBridge-Channel (0x8946)   |  0x0  |Channel Protocol=0x009 |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |          Flags        |  ERR  |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
Address Flush Protocol Specific:
   +-+-+-+-+-+-+-+-+
   | K-nicks       |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | Nickname 1                    | Nickname 2                    |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | Nickname ...                  | Nickname K-nicks              |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | 0             |  TLVs ...
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...
```

Figure 3: Address Flush Message - Extensible Case

Channel Protocol, K-nicks, Nickname: These fields are as specified in Section 2.1.

TLVs: If the byte immediately before the TLVs field, which is the byte labeled "K-VLBs" in Figure 2, is zero, as shown in Figure 3, the remainder of the message consists of TLVs encoded as shown in Figure 4.

```
 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
|  Type         |  Length       |  Value
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
```

Figure 4: Type, Length, Value

Type: The 8-bit TLV type as shown in the table below. See subsections of Section 2.2 for details on each type assigned below. If the type is reserved or not known by a receiving RBridge, that receiving RBridge ignores the value and skips to the next TLV by use of the Length byte. There is no provision for a list of VLAN ID TLVs as there are few enough of them that an arbitrary subset of VLAN IDs can be represented as a bit map.

Type	Description	Reference
0	Reserved	[RFC8383]
1	Blocks of VLANs	[RFC8383]
2	Bit Map of VLANs	[RFC8383]
3	Blocks of FGLs	[RFC8383]
4	List of FGLs	[RFC8383]
5	Bit Map of FGLs	[RFC8383]
6	All Data Labels	[RFC8383]
7	MAC Address List	[RFC8383]
8	MAC Address Blocks	[RFC8383]
9-254	Unassigned	
255	Reserved	[RFC8383]

Length: The 8-bit unsigned integer length in bytes of the remaining information in the TLV after the Length byte. The Length MUST NOT imply that the value extends beyond the end of the RBridge Channel-Protocol-Specific Payload area. If it does, the Address Flush message is corrupt and MUST be ignored.

Value: Depends on the TLV type.

In an extensible Address Flush message, when the TLVs are parsed, those TLVs having unknown types are ignored by the receiving RBridge. There may be multiple instances of TLVs with the same Type in the same Address Flush message, and TLVs are not required to be in any particular order.

- All RBridges implementing the Address Flush RBridge Channel protocol message MUST implement types 1 and 2, the VLAN types, and Type 6, which indicates addresses are to be flushed for all Data Labels.

- RBridges that implement the Address Flush message and implement FGL ingress/egress MUST implement types 3, 4, and 5, the FGL types. (An RBridge that is merely FGL safe [RFC7172], but cannot egress FGL TRILL Data packets, SHOULD ignore the FGL types, as it will not learn any FGL-scoped MAC addresses from the data plane.)

- RBridges that implement the Address Flush message SHOULD implement types 7 and 8 so that specific MAC addresses can be flushed. If they do not, the effect will be to flush all MAC addresses for the indicated Data Labels, which may be inefficient as any MAC addresses not intended to be flushed will have to be relearned.

The parsing of the TLVs by a receiving RBridge results in three pieces of information:

1. a flag indicating whether one or more Type 6 TLVs (All Data Labels) were encountered;

2. a set of Data Labels accumulated from VLAN and/or FGL specifying TLVs in the message; and,

3. if the MAC address TLV types are implemented, a set of MAC addresses accumulated from MAC-address-specifying TLVs in the message.

VLANs/FGLs might be indicated more than once due to overlapping blocks or the like, and a VLAN/FGL is included in the above set of VLANs/FGLs if it occurs in any TLV in the Address Flush message. A MAC address might be indicated more than once due to overlapping blocks or the like, and a particular MAC address is included in the above set of MAC addresses if it occurs in any TLV in the Address Flush message.

After the above information has been accumulated by parsing the TLVs, three sets are derived as described below: a set of nicknames, a set of Data Labels, and a set of MAC addresses. The address flush operation at the receiver applies to the cross product of these derived sets. That is, a { Data Label, MAC address, nickname } triple is flushed if and only if the Data Label matches an element in the derived set of Data Labels, the MAC address matches an element in the derived set of MAC address, and the nickname matches an element in the derived set of nicknames. In the case of Data Labels and MAC addresses, a special value of the set, {ALL}, is permitted, which matches all values.

The sets are derived as follows:

Data Labels set:
- If the Type 6 TLV has been encountered, the set is {ALL}, else,
- if any Data Labels have been accumulated by processing Data Label TLVs (Types 1, 2, 3, 4, and 5), the set is those accumulated Data Labels, else,
- the Data Labels set is null and the Address Flush message does nothing.

MAC Addresses set:
- In the receiver does not implement the MAC address types (Types 7 and 8) or it does implement those types but no MAC addresses are accumulated in parsing the TLVs, then the MAC Address set is {ALL},
- else, the MAC Addresses set is the set of MAC addresses accumulated in processing the TLVs.

Nicknames set:
- If the K-nicks field in the Address Flush message was zero, then the ingress nickname in the TRILL Header of the message is the sole nickname set member, else,
- the nicknames set members are the K-nicks nicknames listed in the Address Flush message.

The various formats below are provided for encoding efficiency. A block of values is most efficient when there are a number of consecutive values. A bit map is most efficient if there are scattered values within a limited range. And a list of single values is most efficient if there are widely scattered values.

2.2.1. Blocks of VLANs

If the TLV Type is 1, the value is a list of blocks of VLANs as follows:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Type = 1      | Length        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| RESV  | Start.VLAN 1          | RESV  | End.VLAN 1            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| RESV  | Start.VLAN 2          | RESV  | End.VLAN 2            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| RESV  | Start.VLAN ...        | RESV  | End.VLAN ...          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The meaning of Start.VLAN and End.VLAN is as specified in Section 2.1. Length MUST be a multiple of 4. If Length is not a multiple of 4, the TLV is corrupt and the Address Flush message MUST be discarded.

2.2.2. Bit Map of VLANs

If the TLV Type is 2, the value is a bit map of VLANs as follows:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Type = 2      | Length        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
| RESV  | Start.VLAN            | Bits...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
```

The value portion of the TLV begins with two bytes having the 12-bit starting VLAN ID right justified (the top 4 bits are as specified in Section 2.1 RESV). This is followed by bytes with one bit per VLAN ID. The high order bit of the first byte is for VLAN N. The next-to-the-highest order bit is for VLAN N+1. The low order bit of the first byte is for VLAN N+7. The high order bit of the second byte, if there is a second byte, is for VLAN N+8, and so on. If that bit is a one, the Address Flush message applies to that VLAN. If that bit is a zero, then addresses that have been learned in that VLAN are not flushed. Note that Length MUST be at least 2. If Length is 0 or 1, the TLV is corrupt and the Address Flush message MUST be discarded. VLAN IDs do not wrap around. If there are enough bytes so that some bits correspond to VLAN ID 0xFFF or higher, those bits are ignored, but the message is still processed for bits corresponding to valid VLAN IDs.

2.2.3. Blocks of FGLs

If the TLV Type is 3, the value is a list of blocks of FGLs as follows:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Type = 3      | Length        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Start.FGL 1                                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| End.FGL 1                                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Start.FGL 2                                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| End.FGL 2                                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Start.FGL ...                                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| End.FGL ...                                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The TLV value consists of sets of Start.FGL and End.FGL numbers. The Address Flush information applies to the FGLs in that range, inclusive. A single FGL is indicated by setting both Start.FGL and End.FGL to the same value. If End.FGL is less than Start.FGL, considering them as unsigned integers, that block is ignored, but the Address Flush message is still processed for any other blocks present. For this Type, Length MUST be a multiple of 6; if it is not, the TLV is corrupt and the Address Flush message MUST be discarded if the receiving RBridge implements Type 3.

2.2.4. list of FGLs

If the TLV Type is 4, the value is a list of FGLs as follows:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Type = 4      | Length        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| FGL 1                                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| FGL 2                                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| FGL ...                                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The TLV value consists of FGL numbers each in 3 bytes. The Address Flush message applies to those FGLs. For this Type, Length MUST be a multiple of 3; if it is not, the TLV is corrupt and the Address Flush message MUST be discarded if the receiving RBridge implements Type 4.

2.2.5. Big Map of FGLs

If the TLV Type is 5, the value is a bit map of FGLs as follows:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Type = 5      | Length        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Start.FGL                                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Bits...
+-+-+-+-+-+-+-+-
```

The TLV value consists of three bytes with the 24-bit starting FGL value N. This is followed by bytes with one bit per FGL. The high order bit of the first byte is for FGL N. The next-to-the-highest order bit is for FGL N+1. The low order bit of the first byte is for FGL N+7. The high order bit of the second byte, if there is a second byte, is for FGL N+8, and so on. If that bit is a one, the Address Flush message applies to that FGL. If that bit is a zero, then addresses that have been learned in that FGL are not flushed. Note that Length MUST be at least 3. If Length is 0, 1, or 2 for a Type 5 TLV, the TLV is corrupt and the Address Flush message MUST be discarded if Type 5 is implemented. FGLs do not wrap around. If there are enough bytes so that some bits correspond to an FGL higher than 0xFFFFFF, those bits are ignored, but the message is still processed for bits corresponding to valid FGLs.

2.2.6. All Data Labels

If the TLV Type is 6, the value is null as follows:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Type = 6      | Length = 0    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

This type is used when an RBridge wants to withdraw all addresses for all the Data Labels (all VLANs and FGLs). Length MUST be zero. If Length is any other value, the TLV is corrupt and the Address Flush message MUST be discarded.

2.2.7. MAC Address List

If the TLV Type is 7, the value is a list of MAC addresses as follows:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Type = 7      | Length        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| MAC 1 upper half                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| MAC 1 lower half                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| MAC 2 upper half                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| MAC 2 lower half                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| MAC ... upper half                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| MAC ... lower half                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The TLV value consists of a list of 48-bit MAC addresses. Length MUST be a multiple of 6. If it is not, the TLV is corrupt, and the Address Flush message MUST be discarded if the receiving RBridge implements Type 7.

2.2.8. MAC Address Blocks

If the TLV Type is 8, the value is a list of blocks of MAC addresses as follows:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Type = 8      | Length        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| MAC.start 1 upper half                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| MAC.start 1 lower half                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| MAC.end 1 upper half                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| MAC.end 1 lower half                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| MAC.start 2 upper half                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| MAC.start 2 lower half                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| MAC.end 2 upper half                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| MAC.end 2 lower half                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| MAC.start ... upper half                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| MAC.start ... lower half                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| MAC.end ... upper half                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| MAC.end ... lower half                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The TLV value consists of sets of Start.MAC and End.MAC numbers. The Address Flush information applies to the 48-bit MAC Addresses in that range, inclusive. A single MAC address is indicated by setting both Start.MAC and End.MAC to the same value. If End.MAC is less than Start.MAC, considering them as unsigned integers, that block is ignored but the Address Flush message is still processed for any other blocks present. For this Type, Length MUST be a multiple of 12; if it is not, the TLV is corrupt and the Address Flush message MUST be discarded if the receiving RBridge implements Type 7.

3. IANA Considerations

3.1. Address Flush RBridge Channel Protocol Number

IANA has assigned 0x009 as the Address Flush RBridge Channel Protocol number from the range of RBridge Channel protocols allocated by Standards Action [RFC7178] [RFC8126].

The added entry to the "RBridge Channel Protocols" registry at <https://www.iana.org/assignments/trill-parameters/> is as follows:

Protocol	Description	Reference
0x009	Address Flush	[RFC8383]

3.2. TRILL Address Flush TLV Types

IANA has created the "TRILL Address Flush TLV Types" registry at <https://www.iana.org/assignments/trill-parameters/> as a subregistry of the "RBridge Channel Protocols" registry. Registry headers are as below. The initial entries are as in the table in Section 2.2.

Registry: TRILL Address Flush TLV Types
Registration Procedures: IETF Review
Reference: [RFC8383]

4. Security Considerations

The Address Flush RBridge Channel Protocol itself provides no security assurances or features. However, Address Flush protocol messages can be secured by use of the RBridge Channel Header Extension [RFC7978]. It is RECOMMENDED that all RBridges that implement the Address Flush message be configured to ignore such messages unless they have been secured with an RBridge Channel Header Extension that meets local security policy.

If RBridges receiving Address Flush messages do not require them to be at least authenticated, they are relatively easy to forge. In that case, such forged Address Flush messages can reduce network efficiency, by purging useful learned information that will have to be relearned. This provides a denial-of-service attack, but cannot cause incorrect operation in the sense that it cannot cause a frame to be improperly delivered.

See [RFC7178] for general RBridge Channel Security Considerations.

See [RFC6325] for general TRILL Security Considerations.

5. References

5.1. Normative References

[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>.

[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>.

[RFC7178] Eastlake 3rd, D., Manral, V., Li, Y., Aldrin, S., and D. Ward, "Transparent Interconnection of Lots of Links (TRILL): RBridge Channel Support", RFC 7178, DOI 10.17487/RFC7178, May 2014, <https://www.rfc-editor.org/info/rfc7178>.

[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>.

[RFC7978] Eastlake 3rd, D., Umair, M., and Y. Li, "Transparent Interconnection of Lots of Links (TRILL): RBridge Channel Header Extension", RFC 7978, DOI 10.17487/RFC7978, September 2016, <https://www.rfc-editor.org/info/rfc7978>.

[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>.

5.2. Informative References

[RFC4762] Lasserre, M., Ed. and V. Kompella, Ed., "Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling", RFC 4762, DOI 10.17487/RFC4762, January 2007, <https://www.rfc-editor.org/info/rfc4762>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

Acknowledgements

The following are thanked for their contributions:

Ramkumar Parameswaran, Henning Rogge

Authors' Addresses

Weiguo Hao
Huawei Technologies
101 Software Avenue,
Nanjing 210012
China

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

Donald Eastlake 3rd
Huawei Technologies
155 Beaver Street
Milford, MA 01757
United States of America

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

Yizhou Li
Huawei Technologies
101 Software Avenue,
Nanjing 210012
China

Phone: +86-25-56624629
Email: liyizhou@huawei.com

Mohammed Umair
Cisco
Cessna Business Park, Kadubeesanahalli Village, Hobli,
Sarjapur, Varthur Main Road, Marathahalli,
Bengaluru, Karnataka 560087
India

Email: mohammed.umair2@gmail.com