INTERNET-DRAFT
Intended status: Proposed Standard

Donald Eastlake
Yizhou Li
Huawei
Radia Perlman
Intel

Expires: April 19, 2014 October 20, 2013

TRILL: Interface Addresses APPsub-TLV
<draft-eastlake-trill-ia-appsubtlv-03.txt>

Abstract

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

Table of Contents

1. Introduction

This document specifies a TRILL (Transparent Interconnection of Lots of Links) [RFC6325] IS-IS application sub-TLV (APPsub-TLV [RFC6823]) that enables the convenient representation of sets of addresses such that all of the addresses in each set designate the same interface (port). For example, an EUI-48 MAC (Extended Unique Identifier 48-bit, Media Access Control [RFC5342bis]) address, IPv4 address, and IPv6 address can be reported as all three designating the same interface. In addition, a Data Label (VLAN or Fine Grained Label (FGL [RFCfgl])) is specified for the interface along with the TRILL switch and, optional the TRILL switch port, from which the interface is reachable. Such information could be use in some cases to synthesize responses to or by-pass the need for the Address Resolution Protocol (ARP [RFC826]), the IPv6 Neighbor Discovery (ND [RFC4861]) protocol, or the flooding of unknown MAC addresses [DirectoryFramework].

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

Although, in some IETF protocols, address field types are represented by Ethertype [RFC5342bis] or Hardware Type [RFC5494], only Address Family Number (AFN) is used in this APPsub-TLV to represent address field type.

1.1 Conventions Used in This Document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

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

AFN: Address Family Number

APPsub-TLV: Application sub-TLV [RFC6823].

Data Label: VLAN or FGL.

FGL: Fine Grained Label [RFCfgl].

IA: Interface Addresses.

RBridge: An alternative name for a TRILL switch.

TRILL switch: A device that implements the TRILL protocol.

2. Format of the Interface Addresses APPsub-TLV

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

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

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

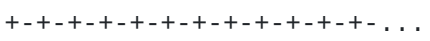


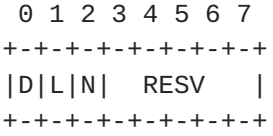
Figure 1. The Interface Addresses APPsub-TLV

- Type: Interface Addresses TRILL APPsub-TLV type, set to TBD[#2 suggested] (IA-SUBTLV).

- Length: Variable, minimum 6, maximum 250 when inside a TRILL GENINFO TLV [ESADI], maximum 255 in unconstrained contexts. If length is 5 or less or if the APPsub-TLV extends beyond an encompassing TRILL GENINFO TLV, the APPsub-TLV MUST be ignored.

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

- Flags: A byte of flags as follows:



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

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

  N: Notify flag: When a TRILL switch receives a new IA APPsub-TLV (one in a ESADI LSP fragment with a higher sequence number or a new message of some other type) and the N bit is one, the TRILL switch then checks the contens of the APPsub-TLV for IP address to MAC address mappings. If an IPv4 to MAC address mapping is found, gratuitous ARPs [RFC826] are sent and if an IPv6 to MAC address mapping is found, spontaneous Neighbor Advertisements [RFC4861] are sent. In both cases, these are sent out all the ports of the TRILL switch that offer end station service and are in the VLAN or FGL of the APPsub-TLV information.

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

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

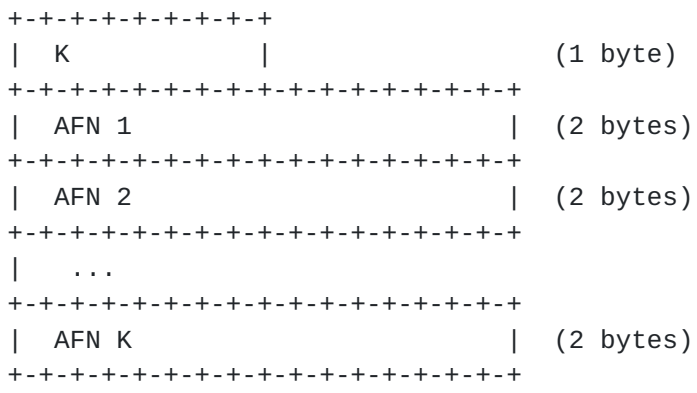
- Addr Sets End: The unsigned offset of the byte, within the IA

APPsub-TLV value part, of the last byte of the last Address Set.

This will be the byte just before the first sub-sub-TLV if any sub-sub-TLVs are present (see Section 3). If this is equal to Size, there are no sub-sub-TLVs. If this is greater than Size, the IA APPsub-TLV is corrupt and MUST be discarded.

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

If the Template uses explicit AFNs, it looks like the following.



For K in the 32 to 103 range, values indicate combinations of a specific number of MAC addresses, IPv4 addresses, IPv6 addresses, and TRILL switch port IDs in that order. The value of K is

K = 32 + M + 3*v4 + 9*v6 + 36*P

where M is 0, 1, or 2 (0 if no MAC address is present, 1 if a 48-bit MAC is present, 2 if a MAC/24 (see Section 5.1) is present), v4 is the number of IPv4 addresses (limited to 0, 1, or 2) and v6 is the number of IPv6 addresses (limited to 0 through 3 inclusive), and P is the number of TRILL switch port IDs (limited to 0 or 1). That equation specifies values of K from 32 through 103. Values from 104 through 254 of the byte value are available for assignment by Expert Review (see Section 5). K = 255 indicates a three byte Template field as specified above. All values (0

through 65,545) of this two byte value are available for assignment by Expert Review.

If an unknown Template K value in the range 104 to 254 is received or a K of 255 followed by an unknown two byte value, the IA APPsub-TLV MUST be ignored.

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

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

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

Different IA APPsub-TLVs within the same or different LSPs or other data structures may have different Templates. The same AFN may occur more than once in a Template and the same address may occur in different address sets. For example, an EUI-48 MAC address interface might have three different IPv6 addresses. This could be represented by an IA APPsub-TLV whose Template specifically provided for one EUI-48 address and three IPv6 addresses, which might be an efficient format if there were multiple interfaces with that pattern. Alternatively, a Template with one 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 the Template specifies; however, the presence of the Addr Set End field means that the sub-sub-TLVs, if any, can always be located by a receiver. An RBridge can be assumed to know the size of the AFNs mentioned in Section 5. 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 MUST be ignored.

3. IA APPsub-TLV sub-sub-TLVs

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

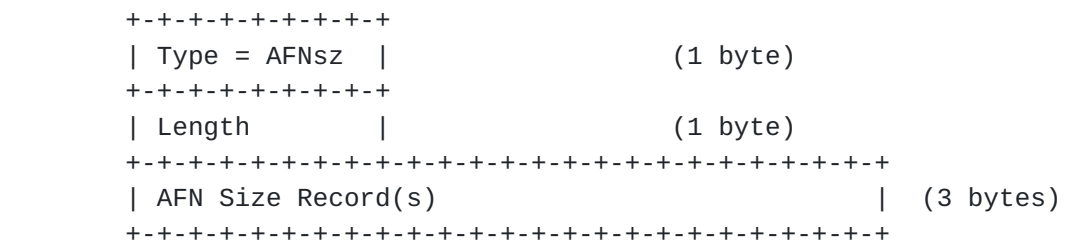
There is no ordering restriction on sub-sub-TLVs. Unless otherwise specified each sub-sub-TLV type can occur zero, one, or many times in an IA APPsub-TLV.

3.1 AFN Size sub-sub-TLV

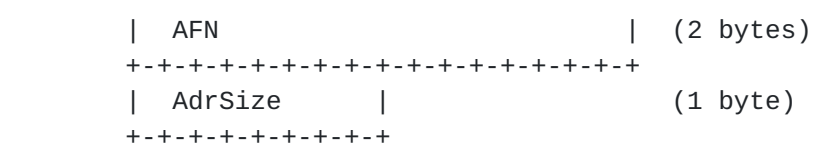
Using this sub-TLV, the originating RBridge can specify the size of an address type. This is useful under two circumstances as follows:

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, then at least the IA APPsub-TLV can be parsed and possibly other addresses in each address set can still be used.

2. If an AFN occurs in the Template that represents a variable length address, this sub-sub-TLV gives its size for all occurrences in that IA APPsub-TLV. (It is believed that the addresses specified by all currently assigned AFNs are fixed length.)



Where each AFN Size Record is structured as follows:



o Type: AFN-Size sub-sub-TLV type, set to 1 (AFNsz).

o Length: 3*n where n is the number of AFN Size Records present. If Length is not a multiple of 3, the sub-sub-TLV MUST be ignored.

- AFN Size Record(s): Zero or more 3-byte records, each giving the size of an address type identified by an AFN,

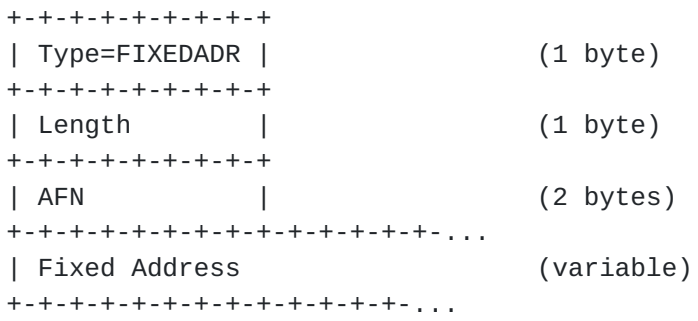
- AFN: The AFN whose length is being specified by the AFN Size Record.

- AdrSize: The length in bytes of addresses specified by the AFN field as an unsigned integer.

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

3.2 Fixed Address sub-sub-TLV

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



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

- Length: variable, minimum 3. If Length is 2 or less, the sub-sub-TLV MUST be ignored.

- AFN: Address Family Number of the Fixed Address.

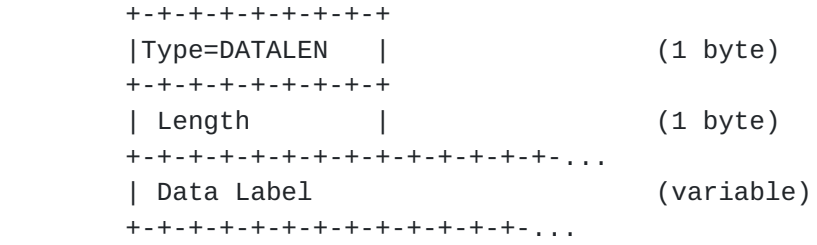
- Fixed Address: The address of the type indicated by the preceding AFN field that is considered to be part of every Address Set in the IA APPsub-TLV.

3.3 Data Label sub-sub-TLV

This sub-sub-TLV indicates the Data Label within which the interfaces

listed in the IA APPsub-TLV are reachable. It is useful if the IA

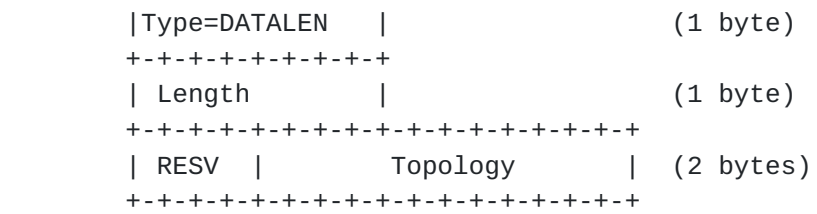
APPsub-TLV occurs outside of the context of an [ESADI] or other type of message specifying the Data Label or if it is desired and permitted to override that specification. Multiple occurrences of this sub-sub-TLV indicate that the interface is reachable in all of the Data Labels given.



- Type: Data Label sub-TLV type, set to 3 (LABEL).
- Length: 2 or 3. If Length is some other value, the sub-sub-TLV is ignored.
- 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].

3.4 Topology sub-sub-TLV

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



- Type: Topology sub-TLV type, set to 4 (TOPOLOGY).
- Length: 2. If Length is some other values, the sub-sub-TLV is ignored.

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

receipt.

- Topology: The 12-bit topology number [RFC5120].

4. Security Considerations

The integrity of address mapping and reachability information and the correctness of Data Labels (VLANs or FLGs [RFCfgl]) are very important. Forged, altered, or incorrect address mapping or Data Labeling can lead to delivery of packets to the incorrect party, violating security policy. However, this document merely describes a data format and does not provide any explicit mechanisms for securing that information, other than a few trivial consistency checks that might detect some corrupted data. Security on the wire, or in storage, for this data is to be providing by the transport or storage used. For example, when transported with [ESADI], [ESADI] security mechanisms can be used.

The address mapping and reachability information, if known to be complete and correct, can be used to detect some cases of forged packet source addresses [DirectoryFramework]. In particular, if native traffic is received by a TRILL switch that would otherwise accept it but authoritative data indicates the source address should not be reachable from the receiving TRILL switch, that traffic should be discarded. The data format specified in this document may optionally include RBridge Port ID number so that this forged address filtering can be optionally applied with port granularity.

See [RFC6325] for general TRILL Security Considerations.

5. IANA Considerations

As specified below, IANA has allocated new AFN numbers and IANA is requested create the TRILL IS-APPsub-TLV sub-sub-TLV subregistry.

5.1 Additional AFN Number Allocation

IANA has alocated AFN numbers as follows:

Number	Description	References
16391	OUI	This document.
16392	MAC/24	This document.
16393	MAC/40	This document.
16394	IPv6/64	This document.
16395	RBridge Port ID	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 or MAC/40 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. An OUI provided by a Fixed Address sub-sub-TLV is ignored if the IA APPsub-TLV has no MAC/24 or MAC/40 in its template.

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

MAC/40 is a suffix as specified above except that it is 40-bit so the result of combining it with an OUI is a 64-bit MAC address.

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-48 or 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 [RFC5342bis].

Other AFNs can be found at http://www.iana.org/assignments/address-family-numbers

The following already allocated AFN values may be particularly useful for IA APPsub-TLVs:

Hex	Decimal	Description	References
0001	1	IPv4	
0002	2	IPv6	
4005	16,389	48-bit MAC	[RFC5342bis]
4006	16,390	64-bit MAC	[RFC5342bis]

5.2 IA APPsub-TLV Sub-Sub-TLVs SubRegistry

IANA is requested to establish a new subregistry of the TRILL Parameter Registry for sub-sub-TLVs of the Interface Addresses APPsub-TLV with initial contents as shown below.

Name: Interface Addresses APPsub-TLV Sub-Sub-TLVs

Procedure: Expert Review

Reference: This document

Type	Description	Reference
0	Reserved	
1	AFN Size	This document
2	Fixed Address	This document
3	Data Label	This document
4	Topology	This document
5-254	Available	
255	Reserved	

Acknowledgments

The authors gratefully acknowledge the contributions and review by the following:

Linda Dunbar

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

Appendix A: Examples

Below are example IA APPsub-TLVs.

A.1 Simple Example

Below is an annotated IA APPsub-TLV carrying two simple pairs of EUI-48 MAC addresses and IPv4 addresses from a Push Directory [DirectoryFramework]. No sub-sub-TLVs are included.

```
0x02(TBD)     Type: Interface Addresses
26            Size: 26 (=0x1A)
0x1234        RBridge Nickname from which reachable
0b10000000    Flags: Push Directory data
0xE3          Confidence
26            Address Sets End: 26 (=0x1A)
35            Template: 35 (0x23) = 32 + 1(MAC48) + 3*1(IPv4)

      Address Set One
0x00005E0053A9   48-bitMAC address
198.51.100.23    IPv4 address

      Address Set Two
0x00005E00536B   48-bit MAC address
203.0.113.201    IPv4 address
```

Size includes 6 for the fixed fields though and including the one byte template, plus 2 times the Address Set size. Each Address Set is 10 bytes, 6 for the 48-bit MAC address plus 4 for the IPv4 address. So total size is 6 + 2*10 = 26.

See Section 2 for more information on Template.

A.2 Complex Example

Below is an annotated IA APPsub-TLV carrying three sets of addresses, each consisting of an EUI-48 MAC address, an IPv4 addresses, an IPv6 address, and an RBridge Port ID, all from a Push Directory [DirectoryFramework]. The IPv6 address for each address set is synthesized from the MAC address given in that set and the IPv6/64 64-bit prefix provided through a Fixed Address sub-sub-TLV. In addition, a sub-sub-TLV is included that provides an FGL which overrides whatever Data Label may be provided by the envelope (for example [ESADI]) within which this IA APPsub-TLV occurs.

```
    0x02(TBD)     Type: Interface Addresses
    59            Size: 59 (=0x3B)
    0x4321        RBridge Nickname from which reachable
    0b10000000    Flags: Push Directory data
    0xD3          Confidence
    42            Address Sets End: 42 (=0x2A)
    72            Template: 72(0x48)=32+1(MAC48)+3*1(IPv4)+36*1(P)

          Address Set One
    0x00005E0053DE   48-bitMAC address
    198.51.100.105   IPv4 address
    0x1DE3           RBridge Port ID

          Address Set Two
    0x00005E0053E3   48-bit MAC address
    203.0.113.89     IPv4 address
    0x1DEE           RBridge Port ID

          Address Set Three
    0x00005E0053D3   48-bit MAC address
    192.0.2.139      IPv4 address
    0x01DE           RBridge Port ID

          sub-sub-TLV One
    0x03             Type: Data Label
    0x03             Length: implies FGL
    0xD3E3E3         Fine Grained Label

          sub-sub-TLV Two
    0x02             Type: Fixed Address
    0x0A             Size: 0x0A = 10
    0x400A           AFN: IPv6/64
    0x0x20010DB800000000   IPv6 Prefix: 2001:DB8::
```

See Section 2 for more information on Template.

The Fixed Address sub-sub-TLV causes the IPv6/64 value give to be treated as if it occurred as a 4th entry inside each of the three Address Sets. When there is an IPv6/64 entry and a 48-bit MAC entry, the MAC value is expanded by inserting 0xFFFE immediately after the OUI and the resulting 64-bit value is used as the lower 64 bits of the resulting IPv6 address [RFC5342bis]. As a result, a receiving TRILL switch would treat the three Address Sets shown as if they had an IPv6 address in them as follows:

```
      Address Set One
0x20010DB80000000000005EFFFE0053DE  IPv6 Address

      Address Set Two
0x20010DB80000000000005EFFFE0053E3  IPv6 Address

      Address Set Three
0x20010DB80000000000005EFFFE0053D3  IPv6 Address
```

As an alternative to the compact "well know value" Template encoding used in this example above, the less compact explicit AFN encoding could have been used. In that case, the IA APPsub-TLV would have started as follows:

```
0x02(TBD)     Type: Interface Addresses
65            Size: 65 (=0x41)
0x4321        RBridge Nickname from which reachable
0b10000000    Flags: Push Directory data
0xD3          Confidence
48            Address Sets End: 48 (=0x30)
0x3           Template: 3 AFNs
0x4005        AFN: 48-bit MAC
0x0001        AFN: IPv4
0x400B        AFN: RBridge Port ID
```

As a final point, since the 48-bit MAC addresses in these three Address Sets all have the same OUI (the IANA OUI [RFC5342bis]), it would have been possible to just have a MAC/24 value giving the lower 24 bits of the MAC in each Address Set. The OUI would them be supplied by a second Fixed Address sub-sub-TLV proving the OUI. With N Address Sets, this would have saved 3*N or 9 bytes in this case at the cost of 7 bytes (1 each for the type and length of the sub-sub-TLV, 2 for the OUI AFN number, and 3 for the OUI). So, even with just three Address Sets, there would be a small net saving of 2 bytes. The savings would grow with a larger number of Address Sets.

Normative References

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

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

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

[RFC5120] - Przygienda, T., Shen, N., and N. Sheth, "M-ISIS: Multi Topology (MT) Routing in Intermediate System to Intermediate Systems (IS-ISs)", RFC 5120, February 2008.

[RFC5305] - Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, October 2008.

[RFC5342bis] - Eastlake 3rd, D., "IANA Considerations and IETF Protocol Usage for IEEE 802 Parameters", BCP 141, RFC 5342, September 2008.

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

[RFC6823] - Ginsberg, L., Previdi, S., and M. Shand, "Advertising Generic Information in IS-IS", RFC 6823, December 2012.

[RFCfgl] - D. Eastlake, M. Zhang, P. Agarwal, R. Perlman, D. Dutt, "TRILL: Fine-Grained Labeling", draft-ietf-trill-fine-labeling-07.txt, in RFC Editor's queue.

Informational References

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

[DirectoryFramework] - Dunbar, L., D. Eastlake, R. Perlman, I. Gashinsky, "TRILL Edge Directory Assistance Framework", draft-ietf-trill-directory-framework-07.txt, in RFC Editor's queue.

[DirectoryScheme] - Dunbar, L., D. Eastlake, R. Perlman, I. Gashinsky, Y. Li, "TRILL": Directory Assistance Mechanisms", draft-dunbar-trill-scheme-for-directory-assist, work in progress.

[ESADI] - Zhai, H., F. Hu, R. Perlman, D. Eastlake, O. Stokes, "TRILL (Transparent Interconnection of Lots of Links): The ESADI (End Station Address Distribution Information) Protocol", draft-ietf-trill-esadi-03.txt, work in progress.

[RFC5494] - Arkko, J. and C. Pignataro, "IANA Allocation Guidelines for the Address Resolution Protocol (ARP)", RFC 5494, April 2009.

Authors' Addresses

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

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

Yizhou Li
Huawei Technologies
101 Software Avenue,
Nanjing 210012 China

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

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

Phone: +1-408-765-8080
Email: Radia@alum.mit.edu