IDR Working Group
Internet-Draft
Updates: 5512 (if approved)
Intended status: Standards Track
Expires: January 7, 2016

E. Rosen, Ed.
Juniper Networks, Inc.
K. Patel
Cisco Systems
G. Van de Velde
Alcatel-Lucent
July 6, 2015

Using the BGP Tunnel Encapsulation Attribute without the BGP Encapsulation SAFI

draft-rosen-idr-tunnel-encaps-01

Abstract

RFC 5512 defines a BGP Path Attribute known as the "Tunnel Encapsulation Attribute". This attribute allows one to specify a set of tunnels. For each such tunnel, the attribute can provide additional information used to create a tunnel and the corresponding encapsulation header, and can also provide information that aids in choosing whether a particular packet is to be sent through a particular tunnel. RFC 5512 states that the attribute is only carried in BGP UPDATEs that have the "Encapsulation Subsequent Address Family (Encapsulation SAFI)". This document updates RFC 5512 by removing that restriction, and by specifying semantics for the attribute when it is carried in UPDATEs of certain other SAFIs. This document also extends the attribute by enabling it to carry additional information needed to create the encapsulation headers additional tunnel types not mentioned in RFC 5512. Finally, this document also extends the attribute by allowing it to specify a remote tunnel endpoint address for each tunnel.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

[RFC5512] defines a BGP Path Attribute known as the Tunnel Encapsulation attribute. This attribute consists of one or more TLVs. Each TLV identifies a particular type of tunnel. Each TLV also contains one or more sub-TLVs. Some of the sub-TLVs, e.g., the "Encapsulation sub-TLV", contain information that may be used to form the encapsulation header for the specified tunnel type. Other sub-TLVs, e.g., the "color sub-TLV" and the "protocol sub-TLV", contain information that aids in determining whether particular packets should be sent through the tunnel that the TLV identifies.

[RFC5512] only allows the Tunnel Encapsulation attribute to be attached to BGP UPDATE messages that have the "Encapsulation SAFI" (i.e., UPDATE messages with AFI/SAFI 1/7 or 2/7). In an UPDATE of the Encapsulation SAFI, the NLRI is an address of the BGP speaker originating the UPDATE. Consider the following scenario:

- BGP speaker R1 has received and installed UPDATE U;

- UPDATE U's SAFI is the Encapsulation SAFI;

- UPDATE U has the address R2 as its NLRI;

- UPDATE U has a Tunnel Encapsulation attribute.

- R1 has a packet, P, to transmit to destination D;

- R1's best path to D is a BGP route that has R2 as its next hop;

In this scenario, when R1 transmits packet P, it should transmit it to R2 through one of the tunnels specified in U's Tunnel Encapsulation attribute. The IP address of the remote endpoint of each such tunnel is R2. Packet P is known as the tunnel's "payload".

While the ability to specify tunnel information in a BGP UPDATE is useful, the procedures of [RFC5512] have certain limitations:

o The requirement to use the "Encapsulation SAFI" presents an unfortunate operational cost, as each BGP session that may need to

carry tunnel encapsulation information needs to be reconfigured to support the Encapsulation SAFI.

o There is no way to use the Tunnel Encapsulation attribute to specify the remote endpoint address of a given tunnel; [RFC5512] assumes that the remote endpoint of each tunnel is specified as the NLRI of an UPDATE of the Encapsulation-SAFI.

o If the respective best paths to two different address prefixes have the same next hop, [RFC5512] does not provide a straightforward method to associate each prefix with a different tunnel.

In this document we address these deficiencies by:

o Defining a new "Remote Endpoint Address sub-TLV" that can be included in any of the TLVs contained in the Tunnel Encapsulation attribute. This sub-TLV can be used to specify the remote endpoint address of a particular tunnel.

o Allowing the Tunnel Encapsulation attribute to be carried by BGP UPDATEs of additional AFI/SAFIs. Appropriate semantics are provided for this way of using the attribute.

One of the sub-TLVs defined in [RFC5512] is the "Encapsulation sub-TLV". For a given tunnel, the encapsulation sub-TLV specifies some of the information needed to construct the encapsulation header used when sending packets through that tunnel. This document defines encapsulation sub-TLVs for a number of tunnel types not discussed in

[RFC5512]: VXLAN, VXLAN-GRE, NVGRE, GTP, and MPLS-in-GRE. MPLS-in-UDP [RFC7510] is also supported, but an Encapsulation sub-TLV for it is not needed.

Some of the encapsulations mentioned in the previous paragraph need to be further encapsulated inside UDP and/or IP. [RFC5512] provides no way to specify that certain information is to appear in these outer IP and/or UDP encapsulations. This document provides a framework for including such information in the TLVs of the Tunnel Encapsulation attribute.

When the Tunnel Encapsulation attribute is attached to a BGP UPDATE whose AFI/SAFI identifies one of the labeled address families, it is not always obvious whether the label embedded in the NLRI is to appear somewhere in the tunnel encapsulation header (and if so, where), or whether it is to appear in the payload, or whether it can be omitted altogether. This is especially true if the tunnel encapsulation header itself contains a "virtual network identifier". This document provides a mechanism that allows one to signal (by

using sub-TLVs of the Tunnel Encapsulation attribute) how one wants to use the embedded label when the tunnel encapsulation has its own virtual network identifier field.

[RFC5512] defines a Tunnel Encapsulation Extended Community, that can be used instead of the Tunnel Encapsulation attribute under certain circumstances. This document addresses the issue of how to handle a BGP UPDATE that carries both a Tunnel Encapsulation attribute and one or more Tunnel Encapsulation Extended Communities.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL", when and only when appearing in all capital letters, are to be interpreted as described in [RFC2119].

2. Tunnel Encapsulation Attribute Sub-TLVs

[RFC5512] specifies three sub-TLVs for the Tunnel Encapsulation attribute: the Encapsulation sub-TLV, the Color sub-TLV, and the Protocol Type sub-TLV. In this section we specify a number of additional sub-TLVs. We also specify Encapsulation sub-TLVs for a number of tunnel types that are not mentioned in [RFC5512].

2.1. The Remote Endpoint Sub-TLV

The Remote Endpoint sub-TLV is a sub-TLV whose value field contains three sub-fields:

1. a four-octet Autonomous System (AS) number sub-field

2. a two-octet Address Family sub-field

3. an address sub-field, whose length depends upon the Address Family.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 Autonomous System Number                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Address Family         |          Address                ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               +
~                                                               ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: Remote Endpoint Sub-TLV Value Field

The Address Family subfield contains a value from IANA's "Address Family Numbers" registry. In this document, we assume that the Address Family is either IPv4 or IPv6; use of other address families is outside the scope of this document.

If the Address Family subfield contains the value for IPv4, the address subfield must contain an IPv4 address (a /32 IPv4 prefix). In this case, the length field of Remote Endpoint sub-TLV must contain the value 10 (0xa). IPv4 broadcast addresses are not valid values of this field.

If the Address Family subfield contains the value for IPv6, the address sub-field must contain an IPv6 address (a /128 IPv6 prefix). In this case, the length field of Remote Endpoint sub-TLV must contain the value 22 (0x16). IPv6 link local addresses are not valid values of the IP address field.

In a given BGP UPDATE, the address family (IPv4 or IPv6) of a Remote Endpoint sub-TLV is independent of the address family of the UPDATE itself. For example, an UPDATE whose NLRI is an IPv4 address may have a Tunnel Encapsulation attribute containing Remote Endpoint sub-TLVs that contain IPv6 addresses. Also, different tunnels represented in the Tunnel Encapsulation attribute may have Remote Endpoints of different address families.

A two-octet AS number can be carried in the AS number field by setting the two high order octets to zero, and carrying the number in the two low order octets of the field.

The AS number in the sub-TLV MUST be the number of the AS to which the IP address in the sub-TLV belongs.

There is one special case: the Remote Endpoint sub-TLV MAY have a value field whose Address Family subfield contains 0. This means that the tunnel's remote endpoint is the UPDATE's BGP next hop. If the Address Family subfield contains 0, the Address subfield is omitted, and the Autonomus System number field is set to 0.

If any of the following conditions hold, the Remote Endpoint sub-TLV is considered to be "malformed":

o The sub-TLV contains the value for IPv4 in its Address Family subfield, but the length of the sub-TLV's value field is other than 10 (0xa).

o The sub-TLV contains the value for IPv6 in its Address Family subfield, but the length of the sub-TLV's value field is other than 22 (0x16).

o The sub-TLV contains the value zero in its Address Family field, but the length of the sub-TLV's value field is other than 6, or the Autonomous System subfield is not set to zero.

o The IP address in the sub-TLV's address subfield is not a valid IP address (e.g., it's an IPv4 broadcast address).

o It can be determined that the IP address in the sub-TLV's address subfield does not belong to the non-zero AS whose number is in the its Autonomous System subfield. (See section Section 11 for discussion of one way to determine this.)

If the Remote Endpoint sub-TLV is malformed, the TLV containing it is also considered to be malformed, and the entire TLV MUST be ignored. However, the Tunnel Encapsulation attribute SHOULD NOT be considered to be malformed in this case; other TLVs in the attribute SHOULD be processed (if they can be parsed correctly).

When redistributing a route that is carrying a Tunnel Encapsulation attribute containing a TLV that itself contains a malformed Remote Endpoint sub-TLV, the TLV SHOULD be removed from the attribute before redistribution.

See Section 9 for further discussion of how to handle errors that are encountered when parsing the Tunnel Encapsulation attribute.

If the Remote Endpoint sub-TLV contains an IPv4 or IPv6 address that is valid but not reachable, the sub-TLV is NOT considered to be malformed, and the containing TLV SHOULD NOT be removed from the attribute before redistribution. However, the tunnel identified by the TLV containing that sub-TLV cannot be used until such time as the address becomes reachable. See Section 3.

2.2. Encapsulation Sub-TLVs for Particular Tunnel Types

Tunnel Encapsulation sub-TLVs for the following tunnel types are defined in [RFC5512]: L2TPv3, and GRE.

This section defines Tunnel Encapsulation sub-TLVs for the following tunnel types: VXLAN ([RFC7348]), VXLAN-GPE ([VXLAN-GPE]), NVGRE ([NVGRE]), GTP [GTP-U], and MPLS-in-GRE ([RFC2784], [RFC2890], [RFC4023]).

Rules for forming the encapsulation based on the information in a given TLV are given in Section 7.

2.2.1. VXLAN

This document defines an encapsulation sub-TLV for VXLAN tunnels. When the tunnel type is VXLAN, the following is the structure of the value field in the encapsulation sub-TLV:

```
 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|M|R|R|R|R|R|R|          VN-ID (3 Octets)                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 MAC Address (4 Octets)                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  MAC Address (2 Octets)      |   Reserved                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: VXLAN Encapsulation Sub-TLV

V: This bit is set to 1 to indicate that a valid VN-ID is present in the encapsulation sub-TLV.

M: This bit is set to 1 to indicate that a valid MAC Address is present in the encapsulation sub-TLV.

R: The remaining bits in the 8-bit flags field are reserved for further use. They SHOULD always be set to 0.

VN-ID: If the V bit is set, the VN-id field contains a 3 octet VN-ID value. If the V bit is not set, the VN-id field SHOULD be set to zero.

MAC Address: If the M bit is set, this field contains a 6 octet Ethernet MAC address. If the M bit is not set, this field SHOULD be set to all zeroes.

When forming the VXLAN encapsulation header:

- o The values of the V, M, and R bits are NOT copied into the flags field of the VXLAN header. The flags field of the VXLAN header is set as per [RFC7348].

- o If the M bit is set, the MAC Address is copied into the Inner Destination MAC Address field of the Inner Ethernet Header (see section 5 of [RFC7348]). If the M bit is not set, the Inner Destination MAC address field is set to a configured value. If the M bit is not set, and there is no configured value, the VXLAN tunnel cannot be used.

o See Section 7 to see how the VNI field of the VXLAN encapsulation header is set.

Note that what we are calling a "VXLAN tunnel" is actually an "ethernet-in-VXLAN" tunnel. Although, strictly speaking, VXLAN tunnels only carry ethernet frames, a IP packet or an MPLS packet can be carried through a "VXLAN tunnel" by forming an IP-in-ethernet-in-VXLAN or MPLS-in-ethernet-in-VXLAN tunnel.

2.2.2. VXLAN-GPE

This document defines an encapsulation sub-TLV for VXLAN tunnels. When the tunnel type is VXLAN-GPE, the following is the structure of the value field in the encapsulation sub-TLV:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Ver|V|R|R|R|R|R|                Reserved                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      VN-ID                    |   Reserved    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: VXLAN GPE Encapsulation Sub-TLV

V: This bit is set to 1 to indicate that a valid VN-ID is present in the encapsulation sub-TLV.

R: The bits designated "R" above are reserved for future use. They SHOULD always be set to zero.

Version (Ver): Indicates VXLAN GPE protocol version. If the indicated version is not supported, the TLV that contains this Encapsulation sub-TLV MUST be treated as specifying an unsupported tunnel type. The value of this field will be copied into the corresponding field of the VXLAN encapsulation header.

VN-ID: If the V bit is set, this field contains a 3 octet VN-ID value. If the V bit is not set, this field SHOULD be set to zero.

When forming the VXLAN-GPE encapsulation header:

o The values of the V and R bits are NOT copied into the flags field of the VXLAN-GPE header. However, the values of the Ver bits are copied into the VXLAN-GPE header. Other bits in the flags field of the VXLAN-GPE header are set as per [VXLAN-GPE].

o See Section 7 to see how the VNI field of the VXLAN-GPE encapsulation header is set.

2.2.3. NVGRE

This document defines an encapsulation sub-TLV for NVGRE tunnels. When the tunnel type is NVGRE, the following is the structure of the value field in the encapsulation sub-TLV:

```
 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|M|R|R|R|R|R|R|          VN-ID (3 Octets)                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   MAC Address (4 Octets)                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  MAC Address (2 Octets)      |   Reserved                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 4: NVGRE Encapsulation Sub-TLV

V: This bit is set to 1 to indicate that a valid VN-ID is present in the encapsulation sub-TLV.

M: This bit is set to 1 to indicate that a valid MAC Address is present in the encapsulation sub-TLV.

R: The remaining bits in the 8-bit flags field are reserved for further use. They SHOULD always be set to 0.

VN-ID: If the V bit is set, the VN-id field contains a 3 octet VN-ID value. If the V bit is not set, the VN-id field SHOULD be set to zero.

MAC Address: If the M bit is set, this field contains a 6 octet Ethernet MAC address. If the M bit is not set, this field SHOULD be set to all zeroes.

When forming the NVGRE encapsulation header:

o The values of the V, M, and R bits are NOT copied into the flags field of the NVGRE header. The flags field of the VXLAN header is set as per [NVGRE].

o If the M bit is set, the MAC Address is copied into the Inner Destination MAC Address field of the Inner Ethernet Header (see section 5 of [NVGRE]. If the M bit is not set, the Inner Destination MAC address field is set to a configured value. If

the M bit is not set, and there is no configured value, the NVGRE tunnel cannot be used.

o See Section 7 to see how the VNI field of the VXLAN encapsulation header is set.

2.2.4. GTP

When the tunnel type is GTP [GTP-U], the Encapsulation sub-TLV contains information needed to send data packets through a GTP tunnel, and also contains information needed by the tunnel's remote endpoint to create a "reverse" tunnel back to the transmitter. This allows a bidirectional control connection to be created. The format of the Encapsulation Sub-TLV is:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Remote TEID (4 Octets)                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Local TEID (4 Octets)                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Local Endpoint Address (4/16 Octets (IPv4/IPv6))        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 5: GTP Encapsulation Sub-TLV

Remote TEID: Contains the 32-bit Tunnel Endpoint Identifier of the GTP tunnel through which data packets are to be sent. When data packets are sent through the tunnel, the Remote TEID is carried in the GTP encapsulation header. The GTP header is itself encapsulation within an IP header, whose IP destination address field is set to the value of the Remote Endpoint sub-TLV.

Local TEID: Contains a 32-bit Tunnel Endpoint Identifier of a GTP tunnel assigned by EPC ([vEPC]).

Local Endpoint Address: Contains an IPv4 or IPv6 anycast address. This is used, along with the Local TEID, to set up a tunnel in the reverse direction. See [vEPC] for details.

2.2.5. MPLS-in-GRE

When the tunnel type is MPLS-in-GRE, the following is the structure of the value field in an optional encapsulation sub-TLV:


```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      GRE-Key (4 Octets)                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 6: MPLS-in-GRE Encapsulation Sub-TLV

GRE-Key: 4-octet field [RFC2890] that is generated by the advertising router. The actual method by which the key is obtained is beyond the scope of this document. The key is inserted into the GRE encapsulation header of the payload packets sent by ingress routers to the advertising router. It is intended to be used for identifying extra context information about the received payload. Note that the key is optional. Unless a key value is being advertised, the MPLS-in-GRE encapsulation sub-TLV MUST NOT be present.

Note that the GRE tunnel type defined in [RFC5512] can be used instead of the MPLS-in-GRE tunnel type when it is necessary to encapsulate MPLS in GRE. Including a TLV of the MPLS-in-GRE tunnel type is equivalent to including a TLV of the GRE tunnel type that also includes a Protocol Type sub-TLV ([RFC5512]) specifying MPLS as the protocol to be encapsulated. That is, if a TLV specifies MPLS-in-GRE or if it includes a Protocol Type sub-TLV specifying MPLS, the

GRE tunnel advertised in that TLV MUST NOT be used for carrying IP packets.

2.3. Outer Encapsulation Sub-TLVs

The Encapsulation sub-TLV for a particular tunnel type allows one to specify the values that are to be placed in certain fields of the encapsulation header for that tunnel type. However, some tunnel types require an outer IP encapsulation, and some also require an outer UDP encapsulation. The Encapsulation sub-TLV for a given tunnel type does not usually provide a way to specify values for fields of the outer IP and/or UDP encapsulations. If it is necessary to specify values for fields of the outer encapsulation, additional sub-TLVs must be used. This document defines two such sub-TLVs.

If an outer encapsulation sub-TLV occurs in a TLV for a tunnel type that does not use the corresponding outer encapsulation, the sub-TLV as if it were an unknown type of sub-TLV.

2.3.1. IPv4 DS Field

Most of the tunnel types that can be specified in the Tunnel Encapsulation attribute require an outer IP encapsulation. The IPv4 DS Field sub-TLV can be carried in the TLV of any such tunnel type. It specifies the setting of one-octet Differentiated Services field in the outer IP encapsulation (see [RFC2474]). The value field is always a single octet.

2.3.2. UDP Destination Port

Some of the tunnel types that can be specified in the Tunnel Encapsulation attribute require an outer UDP encapsulation. Generally there is a standard UDP Destination Port value for a particular tunnel type. However, sometimes it is useful to be able to use a non-standard UDP destination port. If a particular tunnel type requires an outer UDP encapsulation, and it is desired to use a UDP destination port other than the standard one, the port to be used can be specified by including a UDP Destination Port sub-TLV. The value field of this sub-TLV is always a two-octet field, containing the port value.

2.4. Embedded Label Handling Sub-TLV

Certain BGP address families (corresponding to particular AFI/SAFI pairs, e.g., 1/4, 2/4, 1/128, 2/128) have MPLS labels embedded in their NLRIs. We will use the term "embedded label" to refer to the MPLS label that is embedded in an NLRI, and the term "labeled address family" to refer to any AFI/SAFI that has embedded labels.

Some of the tunnel types (e.g., VXLAN, VXLAN-GPE, and NVGRE) that can be specified in the Tunnel Encapsulation attribute have an encapsulation header containing "Virtual Network" identifier of some sort. The Encapsulation sub-TLVs for these tunnel types may optionally specify a value for the virtual network identifier.

Suppose a Tunnel Encapsulation attribute is attached to an UPDATE of an embedded address family, and it is decided to use a particular tunnel (specified in one of the attribute's TLVs) for transmitting a packet that is being forwarded according to that UPDATE. When forming the encapsulation header for that packet, different deployment scenarios require different handling of the embedded label and/or the virtual network identifier. The Embedded Label Handling sub-TLV can be used to control the placement of the embedded label and/or the virtual network identifier in the encapsulation.

The Embedded Label Handling sub-TLV may be included in any TLV of the Tunnel Encapsulation attribute. If the Tunnel Encapsulation

attribute is attached to an UPDATE of a non-labeled address family, the sub-TLV is treated as a no-op. If the sub-TLV is contained in a TLV whose tunnel type does not have a virtual network identifier in its encapsulation header, the sub-TLV is treated as a no-op.

The sub-TLV's Length field always contains the value 1, and its value field consists of a single octet. The following values are defined:

1: The payload will be an MPLS packet with the embedded label at the top of its label stack.

2: The embedded label is not carried in the payload, but is carried either in the virtual network identifier field of the encapsulation header, or else is ignored entirely.

Please see Section 7 for the details of how this sub-TLV is used when it is carried by an UPDATE of a labeled address family.

If the Embedded Label sub-TLV is carried by an UPDATE of a non-labeled address family, it is treated as a no-op. However, it SHOULD NOT be stripped from the TLV before the UPDATE is forwarded.

3. Semantics and Usage of the Tunnel Encapsulation attribute

[RFC5512] specifies the use of the Tunnel Encapsulation attribute in BGP UPDATE messages of AFI/SAFI 1/7 and 2/7. That document restricts the use of this attribute to UPDATE messsages of those SAFIs. This document removes that restriction.

The BGP Tunnel Encapsulation attribute MAY be carried in any BGP UPDATE message whose AFI/SAFI is 1/1 (IPv4 Unicast), 2/1 (IPv6 Unicast), 1/4 (IPv4 Labeled Unicast), 2/4 (IPv6 Labeled Unicast), 1/128 (VPN-IPv4 Labeled Unicast), 2/128 (VPN-IPv6 Labeled Unicast), or 25/70 (EVPN). Use of the Tunnel Encapsulation attribute in BGP UPDATE messages of other AFI/SAFIs is outside the scope of this document.

The decision to attach a Tunnel Encapsulation attribute to a given BGP UPDATE is determined by policy. The set of TLVs and sub-TLVs contained in the attribute is also determined by policy.

When the Tunnel Encapsulation attribute is carried in an UPDATE of one of the AFI/SAFIs specifies in the previous paragraph, each TLV MUST have a Remote Endpoint sub-TLV. If a TLV that does not have a Remote Endpoint sub-TLV, that TLV should be treated as if it had a malformed Remote Endpoint sub-TLV (see Section 2.1).

Suppose that:

- o a given packet P must be forwarded by router R;
- o the path along which P is to be forwarded is determined by BGP UPDATE U;

o UPDATE U has a Tunnel Encapsulation attribute, containing at least one TLV that identifies a "feasible tunnel" for packet P. A tunnel is considered feasible if it has the following two properties:

  * The tunnel type is supported (i.e., router R knows how to set up tunnels of that type, how to create the encapsulation header for tunnels of that type, etc.)

  * The tunnel is of a type that can be used to carry packet P (e.g., an MPLS-in-UDP tunnel would not be a feasible tunnel for carrying an IP packet, UNLESS the IP packet can first be converted to an MPLS packet).

  * The tunnel is specified in a TLV whose Remote Endpoint sub-TLV identifies an IP address that is reachable.

Then router R SHOULD send packet P through one of the feasible tunnels identified in the Tunnel Encapsulation attribute of UPDATE U.

If the Tunnel Encapsulation attribute contains several TLVs (i.e., if it specifies several tunnels), router R may choose any one of those tunnels, based upon local policy. If any of tunnels' TLVs contain the Color sub-TLV and/or the Protocol Type sub-TLV defined in [RFC5512], the choice of tunnel may be influenced by these sub-TLVs.

If a particular tunnel is not feasible at some moment because its Remote Endpoint cannot be reached at that moment, the tunnel may become feasible at a later time. When this happens, router R SHOULD reconsider its choice of tunnel to use, and MAY choose to now use the tunnel.

A TLV specifying a non-feasible tunnel is not considered to be malformed or erroneous in any way, and the TLV SHOULD NOT be stripped from the Tunnel Encapsulation attribute before redistribution.

In addition to the sub-TLVs already defined, additional sub-TLVs may be defined that affect the choice of tunnel to be used, or that affect the contents of the tunnel encapsulation header. The documents that define any such additional sub-TLVs must specify the effect that including the sub-TLV is to have.

If it is determined to send a packet through the tunnel specified in a particular TLV of a particular Tunnel Encapsulation attribute, and if that TLV contains a Remote Endpoint sub-TLV, then the tunnel's remote endpoint address is the IP address contained in the sub-TLV. If the TLV does not contain a Remote Endpoint sub-TLV, or if it contains a Remote Endpoint sub-TLV whose value field is all zeroes, then the tunnel's remote endpoint is the IP address specified as the Next Hop of the BGP Update containing the Tunnel Encapsulation attribute.

The procedure for sending a packet through a particular tunnel type to a particular remote endpoint depends upon the tunnel type, and is outside the scope of this document. The contents of the tunnel encapsulation header MAY be influenced by the Encapsulation sub-TLV.

Note that some tunnel types may require the execution of an explicit tunnel setup protocol before they can be used for carrying data. Other tunnel types may not require any tunnel setup protocol. Whenever a new Tunnel Type TLV is defined, the specification of that TLV must describe (or reference) the procedures for creating the encapsulation header used to forward packets through that tunnel type.

If a Tunnel Encapsulation attribute specifies several tunnels, the way in which a router chooses which one to use is a matter of policy, subject to the following constraint: if a router can determine that a given tunnel is not functional, it MUST NOT use that tunnel. In particular, if the tunnel is identified in a TLV that has a Remote Endpoint sub-TLV, and if the IP address specified in the sub-TLV is not reachable from router R, then the tunnel SHOULD be considered non-functional. Other means of determining whether a given tunnel is functional MAY be used; specification of such means is outside the scope of this specification. Of course, if a non-functional tunnel later becomes functional, router R SHOULD reevaluate its choice of tunnels.

If router R determines that it cannot use any of the tunnels specified in the Tunnel Encapsulation attribute, it MAY either drop packet P, or it MAY transmit packet P as it would had the Tunnel Encapsulation attribute not been present. This is a matter of local policy. By default, the packet SHOULD be transmitted as if the Tunnel Encapsulation attribute had not been present.

A Tunnel Encapsulation attribute may contain several TLVs that all specify the same tunnel type. Each TLV should be considered as specifying a different tunnel. Two tunnels of the same type may have different Remote Endpoint sub-TLVs, different Encapsulation sub-TLVs, etc. Choosing between two such tunnels is a matter of local policy.

Once router R has decided to send packet P through a particular tunnel, it encapsulates packet P appropriately and then forwards it according to the route that leads to the tunnel's remote endpoint. This route may itself be a BGP route with a Tunnel Encapsulation attribute. If so, the encapsulated packet is treated as the payload and is encapsulated according to the Tunnel Encapsulation attribute of that route. That is, tunnels may be "stacked".

4. Routing Considerations

4.1. No Impact on BGP Decision Process

The presence of the Tunnel Encapsulation attribute does not affect the BGP bestpath selection algorithm.

Under certain circumstances, this may need to counter-intuitive consequences. For example, suppose:

- o router R1 receives a BGP UPDATE message from router R2, such that
  - * the NLRI of that UPDATE is prefix X,
  - * the UPDATE contains a Tunnel Encapsulation attribute specifying two tunnels, T1 and T2,
  - * R1 cannot use tunnel T1 or tunnel T2, either because the tunnel remote endpoint is not reachable or because R1 does not support that kind of tunnel
- o router R1 receives a BGP UPDATE message from router R3, such that
  - * the NLRI of that UPDATE is prefix X,
  - * the UPDATE contains a Tunnel Encapsulation attribute specifying two tunnels, T3 and T4,
  - * R1 can use at least one of the two tunnels

Since the Tunnel Encapsulation attribute does not affect bestpath selection, R1 may well install the route from R2 rather than the route from R3, even though R2's route contains no usable tunnels.

This possibility must be kept in mind whenever a Remote Endpoint sub-TLV carried by a given UPDATE specifies an IP address that is different than the next hop of that UPDATE.

4.2. Looping, Infinite Stacking, Etc.

Consider a packet destined for address X. Suppose a BGP UPDATE for address prefix X carries a Tunnel Encapsulation attribute that specifies a remote tunnel endpoint of Y. And suppose that a BGP UPDATE for address prefix Y carries a Tunnel Encapsulation attribute that specifies a Remote Endpoint of X. It is easy to see that this will cause an infinite number of encapsulation headers to be put on the given packet.

This could happen as a result of misconfiguration, either accidental or intentional. It could also happen if the Tunnel Encapsulation attribute were altered by a malicious agent. Implementations should be aware of this.

Improper setting (or malicious altering) of the Tunnel Encapsulation attribute could also cause data packets to loop. Suppose a BGP UPDATE for address prefix X carries a Tunnel Encapsulation attribute that specifies a remote tunnel endpoint of Y. Suppose router R receives and processes the update. When router R receives a packet destined for X, it will apply the encapsulation and send the encapsulated packet to Y. Y will decapsulate the packet and forward it further. If Y is further away from X than is router R, it is possible that the path from Y to X will traverse R. This would cause a long-lasting routing loop.

These possibilities must also be kept in mind whenever the Remote Endpoint for a given prefix differs from the BGP next hop for that prefix.

5. Recursive Next Hop Resolution

Suppose that:

o a given packet P must be forwarded by router R1;

o the path along which P is to be forwarded is determined by BGP UPDATE U1;

o UPDATE U1 does not have a Tunnel Encapsulation attribute;

o the next hop of UPDATE U1 is router R2;

o the best path to router R2 is a BGP route that was advertised in UPDATE U2;

o UPDATE U2 has a Tunnel Encapsulation attribute.

Then packet P SHOULD be sent through one of the tunnels identified in the Tunnel Encapsulation attribute of UPDATE U2. See Section 3 for further details.

Note that if UPDATE U1 and UPDATE U2 both have Tunnel Encapsulation attributes, packet P will be carried through a pair of nested tunnels. P will first be encapsulated based on the Tunnel Encapsulation attribute of U1. This encapsulated packet then becomes the payload, and is encapsulated based on the Tunnel Encapsulation attribute of U2. This is another way of "stacking" tunnels (see also Section 3.

6. Tunnel Encapsulation Extended Community

[RFC5512] defines an Encapsulation Extended Community. This Extended Community may be attached to a route any AFI/SAFI to which the Tunnel Encapsulation attribute may be attached. Each such Extended Community identifies a particular tunnel type. If the Encapsulation Extended Community identifies a particular tunnel type, its semantics are exactly equivalent to the semantics of a Tunnel Encapsulation attribute TLV that:

o identifies the same tunnel type, and

o has a Remote Endpoint sub-TLV whose IP address field contains the address of the BGP next hop of the route to which it is attached, and

o has no other sub-TLVs.

7. Use of Virtual Network Identifiers and Embedded Labels when Imposing a Tunnel Encapsulation

Three of the tunnel types that can be specified in a Tunnel Encapsulation TLV have virtual network identifier fields in their encapsulation headers. In the VXLAN and VXLAN-GPE encapsulations, this field is called the VNI field; in the NVGRE encapsulation, this field is called the VSID field.

When one of these tunnel encapsulations is imposed on a packet, the setting of the virtual network identifier field in the encapsulation header depends upon the contents of the Encapsulation sub-TLV (if one is present). When the Tunnel Encapsulation attribute is being carried on a BGP UPDATE of a labeled address family, the setting of the virtual network identifier field also depends upon the contents of the Embedded Label Handling sub-TLV (if present).

This section specifies the procedures for choosing the value to set in the virtual network identifier field of the encapsulation header. These procedures apply only when the tunnel type is VXLAN, VXLAN-GPE, or NVGRE.

7.1. Unlabeled Address Families

This sub-section applies when:

- o the Tunnel Encapsulation attribute is carried on a BGP UPDATE of an unlabeled address family, and

- o at least one of the attribute's TLVs identifies a tunnel type that uses a virtual network identifier, and

- o it has been determined to send a packet through one of those tunnels.

If the TLV identifying the tunnel contains an Encapsulation sub-TLV whose V bit is set, the virtual network identifier field of the encapsulation header is set to the value of the virtual network identifier field of the Encapsulation sub-TLV.

Otherwise, the virtual network identifier field of the encapsulation header is set to a configured value; if there is no configured value, the tunnel cannot be used.

7.2. Labeled Address Families

This sub-section applies when:

- the Tunnel Encapsulation attribute is carried on a BGP UPDATE of a labeled address family, and
- at least one of the attribute's TLVs identifies a tunnel type that uses a virtual network identifier, and
- it has been determined to send a packet through one of those tunnels.

7.2.1. When a Valid VNID has been Signaled

If the TLV identifying the tunnel contains an Encapsulation sub-TLV whose V bit is set, the virtual network identifier field of the encapsulation header is set as follows:

- If the TLV does not contain an Embedded Label Handling sub-TLV, or if it contains an Embedded Label Handling sub-TLV whose value is

  1, then the virtual network identifier field of the encapsulation header is set to the value of the virtual network identifier field of the Encapsulation sub-TLV.

  The embedded label (from the NLRI of the route that is carrying the Tunnel Encapsulation attribute) appears at the top of the MPLS label stack in the encapsulation payload.

- If the TLV contains an Embedded Label Handling sub-TLV whose value is 2, the embedded label is ignored entirely, and the virtual network identifier field of the encapsulation header is set to the value of the virtual network identifier field of the Encapsulation sub-TLV.

7.2.2. When a Valid VNID has not been Signaled

If the TLV identifying the tunnel does not contain an Encapsulation sub-TLV whose V bit is set, the virtual network identifier field of the encapsulation header is set as follows:

- o If the TLV does not contain an Embedded Label Handling sub-TLV, or if it contains an Embedded Label Handling sub-TLV whose value is 1, then the virtual network identifier field of the encapsulation header is set to a configured value.

  If there is no configured value, the tunnel cannot be used.

  The embedded label (from the NLRI of the route that is carrying the Tunnel Encapsulation attribute) appears at the top of the MPLS label stack in the encapsulation payload.

- o If the TLV contains an Embedded Label Handling sub-TLV whose value is 2, the embedded label is copied into the virtual network identifier field of the encapsulation header.

  The embedded label does not appear in the MPLS label stack of the payload.

7.2.3. Applicability Restrictions

In a given UPDATE of a labeled address family, the label embedded in the NLRI is generally a label that is meaningful only to the router whose address appears as the next hop. Certain of the procedures of Section 7.2.1 or Section 7.2.2 cause the embedded label to be carried by a data packet to the router whose address appears in the Remote Endpoint sub-TLV. If the Remote Endpoint sub-TLV does not identify the same router that is the next hop, sending the packet through the

tunnel may cause the label to be misinterpreted at the tunnel's remote endpoint. This may cause misdelivery of the packet.

Therefore the embedded label MUST NOT be carried by a data packet traveling through a tunnel unless it is known that the label will be properly interpreted at the tunnel's remote endpoint. How this is known is outside the scope of this document.

Note that if the Tunnel Encapsulation attribute is attached to a VPN-IP route [RFC4364], and if Inter-AS "option b" (see section 10 of [RFC4364] is being used, and if the Remote Endpoint sub-TLV contains an IP address that is not in same AS as the router receiving the route, it is very likely that the embedded label has been changed. Therefore use of the Tunnel Encapsulation attribute in an "Inter-AS option b" scenario is not supported.

8. Scoping

The Tunnel Encapsulation attribute is defined as a transitive attribute, so that it may be passed along by BGP speakers that do not recognize it. However, it is intended that the Tunnel Encapsulation attribute be used only within a well-defined scope, e.g., within a set of Autonomous Systems that belong to a single administrative entity. If the attribute is distributed beyond its intended scope, packets may be sent through tunnels in a manner that is not intended.

To prevent the Tunnel Encapsulation attribute from being distributed beyond its intended scope, any BGP speaker that understands the attribute MUST be able to filter the attribute from incoming BGP UPDATE messages. When the attribute is filtered from an incoming UPDATE, the attribute is neither processed nor redistributed. This filtering SHOULD be possible on a per-BGP-session basis. For each session, filtering of the attribute on incoming UPDATEs MUST be enabled by default.

In addition, any BGP speaker that understands the attribute MUST be able to filter the attribute from outgoing BGP UPDATE messages. This filtering SHOULD be possible on a per-BGP-session basis. For each session, filtering of the attribute on outgoing UPDATEs MUST be enabled by default.

9. Error Handling

The Tunnel Encapsulation attribute is a sequence of TLVs, each of which is a sequence of sub-TLVs. The final octet of a TLV is determined by its length field. Similarly, the final octet of a sub-TLV is determined by its length field. The final octet of a TLV must also be the final octet of its final sub-TLV. If this is not the case, the TLV MUST be considered malformed. A TLV that is found to be malformed for this reason MUST NOT be processed, and MUST be stripped from the Tunnel Encapsulation attribute before redistribution. Subsequent TLVs in the Tunnel Encapsulation attribute may still be valid, in which case they MUST be processed and redistributed normally.

If a Tunnel Encapsulation attribute does not have any valid TLVs, or it does not have the transitive bit set, the "Attribute Discard" procedure of [ERRORS] is applied.

If a Tunnel Encapsulation attribute can be parsed correctly, but contains a TLV that is not recognized (i.e., the tunnel type is not recognized) by a particular BGP speaker, the attribute is NOT considered to be malformed. The unrecognized TLV MUST be ignored, and the BGP speaker MUST interpret the attribute as if the unrecognized TLV had not been present. If the route carrying the Tunnel Encapsulation attribute is redistributed with the attribute, the unrecognized TLV SHOULD remain in the attribute.

If a TLV of a Tunnel Encapsulation attribute contains a sub-TLV that is not recognized by a particular BGP speaker, the BGP speaker SHOULD process that TLV as if the unrecognized sub-TLV had not been present. If the route carrying the Tunnel Encapsulation attribute is redistributed with the attribute, the unrecognized TLV SHOULD remain in the attribute.

In general, if a TLV contains a sub-TLV that is malformed (e.g., contains a length field whose value is not legal for that sub-TLV), the sub-TLV should be treated as if it were an unrecognized sub-TLV. This document specifies one exception to this rule -- if a TLV contains a malformed Remote Endpoint sub-TLV (as defined in Section 2.1, the entire TLV MUST be ignored, and SHOULD be removed from the Tunnel Encapsulation attribute before the route carrying that attribute is redistributed.

A TLV that does not contain the Remote Endpoint sub-TLV MUST be treated as if it contained a malformed Remote Endpoint sub-TLV.

A TLV identifying a particular tunnel type may contain a sub-TLV that is meaningless for that tunnel type. For example, perhaps the TLV contains a "UDP Destination Port" sub-TLV, but the identified tunnel type does not use UDP encapsulation at all. Sub-TLVs of this sort SHOULD be treated as no-ops. That is, they SHOULD NOT affect the creation of the encapsulation header. However, the sub-TLV MUST NOT be considered to be malformed, and MUST NOT be removed from the TLV before the route carrying the Tunnel Encapsulation attribute is redistributed.

There is no significance to the order in which the TLVs occur within the Tunnel Encapsulation attribute. Multiple TLVs may occur for a given tunnel type; each such TLV is regarded as describing a different tunnel.

10. IANA Considerations

IANA is requested to assign a codepoint from the "BGP Tunnel Encapsulation Attribute Sub-TLVs" registry for "Remote Endpoint", with this document being the reference.

IANA is requested to assign a codepoint from the "BGP Tunnel Encapsulation Attribute Sub-TLVs" registry for "IPv4 DS Field", with this document being the reference.

IANA is requested to assign a codepoint from the "BGP Tunnel Encapsulation Attribute Sub-TLVs" registry for "UDP Destination Port", with this document being the reference.

IANA is requested to assign a codepoint from the "BGP Tunnel Encapsulation Attribute Sub-TLVs" registry for "Embedded Label Handling", with this document being the reference.

IANA is requested to add this document as a reference for tunnel types 8-13 in the "BGP Tunnel Encapsulation Tunnel Types" registry.

11. Security Considerations

The Tunnel Encapsulation attribute can cause traffic to be diverted from its normal path, especially when the Remote Endpoint sub-TLV is used. This can have serious consequences if the attribute is added or modified illegitimately, as it enables traffic to be "hijacked".

The Remote Endpoint sub-TLV contains both an IP address and an AS number. BGP Origin Validation [RFC6811] can be used to obtain assurance that the given IP address belongs to the given AS. While this provides some protection against misconfiguration, it does not prevent a malicious agent from inserting a sub-TLV that will appear valid.

Before sending a packet through the tunnel identified in a particular TLV of a Tunnel Encapsulation attribute, it may be advisable to use

BGP Origin Validation to obtain the following additional assurances:

o the origin AS of the route carrying the Tunnel Encapsulation attribute is correct;

o the origin AS of the route to the IP address specified in the Remote Endpoint sub-TLV is correct, and is the same AS that is specified in the Remote Endpoint sub-TLV.

One then has some level of assurance that the tunneled traffic is going to the same destination AS that it would have gone to had the Tunnel Encapsulation attribute not been present. However, this may not suit all use cases, and in any event is not very strong protection against hijacking.

For these reasons, BGP Origin Validation should not be relied upon exclusively, and the filtering procedures of Section 8 should always be in place.

Increased protection can be obtained by using BGP Path Validation [BGPSEC] to ensure that the route carrying the Tunnel Encapsulation attribute, and the routes to the Remote Endpoint of each specified tunnel, have not been altered illegitimately.

If BGP Origin Validation is used as specified above, and the tunnel specified in a particular TLV of a Tunnel Encapsulation attribute is therefore regarded as "suspicious", that tunnel should not be used. Other tunnels specified in (other TLVs of) the Tunnel Encapsulation attribute may still be used.

12. Acknowledgments

The authors wish to think Ron Bonica, John Drake, Satoru Matushima, Dhananjaya Rao, John Scudder, and Ravi Singh for their review, comments, and/or helpful discussions.

13. Contributor Addresses

Below is a list of other contributing authors in alphabetical order:

Randy Bush
Internet Initiative Japan
5147 Crystal Springs
Bainbridge Island, Washington 98110
United States

Email: randy@psg.com

Robert Raszuk
Mirantis Inc.
615 National Ave. #100
Mountain View, California 94043
United States

Email: robert@raszuk.net

14. References

14.1. Normative References

[ERRORS]   Chen, E., Scudder, J., Mohapatra, P., and K. Patel, "Revised Error Handling for BGP UPDATE Messages", internet-draft draft-ietf-idr-error-handling-19, April 2015.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate

Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC5512] Mohapatra, P. and E. Rosen, "The BGP Encapsulation Subsequent Address Family Identifier (SAFI) and the BGP Tunnel Encapsulation Attribute", RFC 5512, April 2009.

14.2. Informative References

[BGPSEC] Lepinski, M. and S. Turner, "An Overview of BGPsec", internet-draft draft-ietf-sidr-bgpsec-overview, January 2015.

[GTP-U] 3GPP, "GPRS Tunneling Protocol User Plane, TS 29.281", 2014.

[NVGRE] Garg, P. and Y. Wang, "NVGRE: Network Virtualization using Generic Routing Encapsulation", internet-draft draft-sridharan-virtualization-nvgre, April 2015.

[RFC2474] Nichols, K., Blake, S., Baker, F., and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, December 1998.

[RFC2784] Farinacci, D., Li, T., Hanks, S., Meyer, D., and P. Traina, "Generic Routing Encapsulation (GRE)", RFC 2784, March 2000.

[RFC2890] Dommety, G., "Key and Sequence Number Extensions to GRE", RFC 2890, September 2000.

[RFC4023] Worster, T., Rekhter, Y., and E. Rosen, "Encapsulating MPLS in IP or Generic Routing Encapsulation (GRE)", RFC 4023, March 2005.

[RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, February 2006.

[RFC6811] Mohapatra, P., Scudder, J., Ward, D., Bush, R., and R. Austein, "BGP Prefix Origin Validation", RFC 6811, January

2013.

[RFC7348] Mahalingam, M., Dutt, D., Duda, K., Agarwal, P., Kreeger, L., Sridhar, T., Bursell, M., and C. Wright, "Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", RFC 7348, August 2014.

[RFC7510] Xu, X., Sheth, N., Yong, L., Callon, R., and D. Black, "Encapsulating MPLS in UDP", RFC 7510, April 2015.

[vEPC] Matsushima, S. and R. Wakikawa, "Stateless User-Plane Architecture for Virtualized EPC", internet-draft draft-matsushima-stateless-uplane-vepc-04, March 2015.

[VXLAN-GPE] Quinn, P., Manur, R., Kreeger, L., Lewis, D., Maino, F., Smith, M., Agarwal, P., Xu, X., Elzur, U., Garg, P., Melman, D., and R. Manur, "Generic Protocol Extension for VXLAN", internet-draft draft-ietf-nvo3-vxlan-gpe, May 2015.

Authors' Addresses

Eric C. Rosen (editor)
Juniper Networks, Inc.
10 Technology Park Drive
Westford, Massachusetts 01886
United States

Email: erosen@juniper.net

Keyur Patel
Cisco Systems
170 W. Tasman Drive
San Jose, CA 95134
United States

Email: keyupate@cisco.com

Gunter Van de Velde
Alcatel-Lucent
Copernicuslaan 50
Antwerpen 2018
Belgium

Email: gunter.van_de_velde@alcatel-lucent.com