Stream: Internet Engineering Task Force (IETF)
RFC: 9601
Updates: 6040, 2661, 2784, 3931, 4380, 7450
Category: Standards Track
Published: June 2024
ISSN: 2070-1721
Author: B. Briscoe
Independent

RFC 9601
Propagating Explicit Congestion Notification across IP Tunnel Headers Separated by a Shim

Abstract

RFC 6040 on "Tunnelling of Explicit Congestion Notification" made the rules for propagation of Explicit Congestion Notification (ECN) consistent for all forms of IP-in-IP tunnel. This specification updates RFC 6040 to clarify that its scope includes tunnels where two IP headers are separated by at least one shim header that is not sufficient on its own for wide-area packet forwarding. It surveys widely deployed IP tunnelling protocols that use such shim headers and updates the specifications of those that do not mention ECN propagation (including RFCs 2661, 3931, 2784, 4380 and 7450; these RFCs specify L2TPv2, L2TPv3, Generic Routing Encapsulation (GRE), Teredo, and Automatic Multicast Tunneling (AMT), respectively). This specification also updates RFC 6040 with configuration requirements needed to make any legacy tunnel ingress safe.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc9601.

Copyright Notice

Table of Contents

1. Introduction

[RFC6040] on "Tunnelling of Explicit Congestion Notification" made the rules for propagation of Explicit Congestion Notification (ECN) [RFC3168] consistent for all forms of IP-in-IP tunnel.

A common pattern for many tunnelling protocols is to encapsulate an inner IP header (v4 or v6) with one or more shim headers then an outer IP header (v4 or v6). Some of these shim headers are designed as generic encapsulations, so they do not necessarily directly encapsulate an inner IP header. Instead, they can encapsulate headers such as link-layer (L2) protocols that, in turn, often encapsulate IP.

To clear up confusion, this specification clarifies that the scope of [RFC6040] includes any IP-in-IP tunnel, including those with one or more shim headers and other encapsulations between the IP headers. Where necessary, it updates the specifications of the relevant encapsulation protocols with the specific text necessary to comply with [RFC6040].

This specification also updates [RFC6040] to state how operators ought to configure a legacy tunnel ingress to avoid unsafe system configurations.

2. Terminology

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

This specification uses the terminology defined in [RFC6040].

3. Scope of RFC 6040

In Section 1.1 of [RFC6040], its scope is defined as:

> ...ECN field processing at encapsulation and decapsulation for any IP-in-IP tunnelling, whether IPsec or non-IPsec tunnels. It applies irrespective of whether IPv4 or IPv6 is used for either the inner or outer headers.

This was intended to include cases where one or more shim headers sits between the IP headers. Many tunnelling implementers have interpreted the scope of [RFC6040] as it was intended, but it is ambiguous. Therefore, this specification updates [RFC6040] by adding the following scoping text after the sentences quoted above:

> It applies in cases where an outer IP header encapsulates an inner IP header either directly or indirectly by encapsulating other headers that in turn encapsulate (or might encapsulate) an inner IP header.

There is another problem with the scope of [RFC6040]. Like many IETF specifications, [RFC6040] is written as a specification that implementations can choose to claim compliance with. This means it does not cover two important cases:

1. Cases where it is infeasible for an implementation to access an inner IP header when adding or removing an outer IP header
2. Implementations that choose not to propagate ECN between IP headers

However, the ECN field is a non-optional part of the IP header (v4 and v6), so any implementation that creates an outer IP header has to give the ECN field some value. There is only one safe value a tunnel ingress can use if it does not know whether the egress supports propagation of the ECN field; it has to clear the ECN field in any outer IP header to 0b00.

However, an RFC has no jurisdiction over implementations that choose not to comply with it or cannot comply with it, including all those implementations that predated the RFC. Therefore, it would have been unreasonable to add such a requirement to [RFC6040]. Nonetheless, to ensure safe propagation of the ECN field over tunnels, it is reasonable to add requirements on operators to ensure they configure their tunnels safely (where possible). Before stating these configuration requirements in Section 4, the factors that determine whether propagating ECN is feasible or desirable will be briefly introduced.

3.1. Feasibility of ECN Propagation between Tunnel Headers

In many cases, one or more shim headers and an outer IP header are always added to (or removed from) an inner IP packet as part of the same procedure. We call this a tightly coupled shim header. Processing the shim and outer together is often necessary because the shim(s) is not sufficient for packet forwarding in its own right; not unless complemented by an outer header. In these cases, it will often be feasible for an implementation to propagate the ECN field between the IP headers.

In some cases, a tunnel adds an outer IP header and a tightly coupled shim header to an inner header that is not an IP header, but that, in turn, encapsulates an IP header (or might encapsulate an IP header). For instance, an inner Ethernet (or other link-layer) header might encapsulate an inner IP header as its payload. We call this a tightly coupled shim over an encapsulating header.

Digging to arbitrary depths to find an inner IP header within an encapsulation is strictly a layering violation, so it cannot be a required behavior. Nonetheless, some tunnel endpoints already look within a Layer 2 (L2) header for an IP header, for instance, to map the Diffserv codepoint between an encapsulated IP header and an outer IP header [RFC2983]. In such cases at least, it should be feasible to also (independently) propagate the ECN field between the same IP headers. Thus, access to the ECN field within an encapsulating header can be a useful and benign optimization. The guidelines in Section 5 of [RFC9599] give the conditions for this layering violation to be benign.

3.2. Desirability of ECN Propagation between Tunnel Headers

Developers and network operators are encouraged to implement and deploy tunnel endpoints compliant with [RFC6040] (as updated by the present specification) in order to provide the benefits of wider ECN deployment [RFC8087]. Nonetheless, propagation of ECN between IP headers, whether separated by shim headers or not, has to be optional to implement and to use, because:

- legacy implementations of tunnels without any ECN support already exist;
- a network might be designed so that there is usually no bottleneck within the tunnel; and
- if the tunnel endpoints would have to search within an L2 header to find an encapsulated IP header, it might not be worth the potential performance hit.

4. Making a Non-ECN Tunnel Ingress Safe by Configuration

Even when no specific attempt has been made to implement propagation of the ECN field at a tunnel ingress, it ought to be possible for the operator to render a tunnel ingress safe by configuration. The main safety concern is to disable (clear to zero) the ECN capability in the outer IP header at the ingress if the egress of the tunnel does not implement ECN logic to propagate any ECN markings into the packet forwarded beyond the tunnel. Otherwise, the non-ECN egress could discard any ECN marking introduced within the tunnel, which would break all the ECN-based control loops that regulate the traffic load over the tunnel.

Therefore, this specification updates Section 4.3 of [RFC6040] by inserting the following text at the end of the section:

> Whether or not an ingress implementation claims compliance with [RFC6040], [RFC4301], or [RFC3168], when the outer tunnel header is IP (v4 or v6), if possible, the ingress **MUST** be configured to zero the outer ECN field in any of the following cases:
>
> - if it is known that the tunnel egress does not support any of the RFCs that define propagation of the ECN field ([RFC6040], [RFC4301], or the full functionality mode of [RFC3168]);

- if the behaviour of the egress is not known or an egress with unknown behaviour might be dynamically paired with the ingress (one way for an operator of a tunnel ingress to determine the behaviour of an otherwise unknown egress is described in [decap-test]); or
- if an IP header might be encapsulated within a non-IP header that the tunnel ingress is encapsulating, but the ingress does not inspect within the encapsulation.

For the avoidance of doubt, the above only concerns the outer IP header. The ingress **MUST NOT** alter the ECN field of the arriving IP header that will become the inner IP header.

In order that the network operator can comply with the above safety rules, even if an implementation of a tunnel ingress does not claim to support [RFC6040], [RFC4301], or the full functionality mode of [RFC3168]:

- The network operator **MUST NOT** treat the former Type of Service (ToS) octet (IPv4) or the former Traffic Class octet (IPv6) as a single 8-bit field, as the resulting linkage of ECN and Diffserv field propagation between inner and outer is not consistent with the definition of the 6-bit Diffserv field in [RFC2474] and [RFC3260].
- The network operator **SHOULD** be able to be configured to zero the ECN field of the outer header.

For instance, if a tunnel ingress with no ECN-specific logic had a configuration capability to refer to the last 2 bits of the old ToS Byte of the outer (e.g., with a 0x3 mask) and set them to zero, while also being able to allow the DSCP to be re-mapped independently, that would be sufficient to satisfy both implementation requirements above.

There might be concern that the above "**MUST NOT**" makes compliant implementations non-compliant at a stroke. However, by definition, it solely applies to equipment that provides Diffserv configuration. Any such Diffserv equipment that is configuring treatment of the former ToS octet (IPv4) or the former Traffic Class octet (IPv6) as a single 8-bit field must have always been non-compliant with the definition of the 6-bit Diffserv field in [RFC2474] and [RFC3260]. If a tunnel ingress does not have any ECN logic, copying the ECN field as a side effect of copying the DSCP is a seriously unsafe bug that risks breaking the feedback loops that regulate load on a tunnel.

Zeroing the outer ECN field of all packets in all circumstances would be safe, but it would not be sufficient to claim compliance with [RFC6040] because it would not meet the aim of introducing ECN support to tunnels (see Section 4.3 of [RFC6040]).

5. ECN Propagation and Fragmentation/Reassembly

The following requirements update [RFC6040], which omitted handling of the ECN field during fragmentation or reassembly. These changes might alter how many ECN-marked packets are propagated by a tunnel that fragments packets, but this would not raise any backward compatibility issues.

If a tunnel ingress fragments a packet, it **MUST** set the outer ECN field of all the fragments to the same value as it would have set if it had not fragmented the packet.

Section 5.3 of [RFC3168] specifies ECN requirements for reassembly of sets of outer fragments into packets (in outer fragmentation, the fragmentation is visible in the outer header so that the tunnel egress can reassemble the fragments [INTAREA-TUNNELS]). Additionally, the following requirements apply at a tunnel egress:

- During reassembly of outer fragments, the packet **MUST** be discarded if the ECN fields of the outer headers being reassembled into a single packet consist of a mixture of Not ECN-Capable Transport (Not-ECT) and other ECN codepoints.
- If there is mix of ECT(0) and ECT(1) outer fragments, then the reassembled packet **MUST** be set to ECT(1).

  Reasoning: [RFC3168] originally defined ECT(0) and ECT(1) as equivalent, but [RFC3168] has been updated by [RFC8311] to make ECT(1) available for congestion marking differences. The rule is independent of the current experimental use of ECT(1) for Low Latency, Low Loss, and Scalable throughput (L4S) [RFC9331]. The rule is compatible with Pre-Congestion Notification (PCN) [RFC6660], which uses 2 levels of congestion severity, with the ranking of severity from highest to lowest being Congestion Experienced (CE), ECT(1), ECT(0). The decapsulation rules in [RFC6040] take a similar approach.

6. IP-in-IP Tunnels with Tightly Coupled Shim Headers

Below is a list of specifications of encapsulations with tightly coupled shim header(s) in rough chronological order. This list is confined to Standards Track or widely deployed protocols and is not necessarily exhaustive, so for the avoidance of doubt, the scope of [RFC6040] is defined in Section 3.

- Point-to-Point Tunneling Protocol (PPTP) [RFC2637]
- Layer 2 Tunnelling Protocol (L2TP), specifically L2TPv2 [RFC2661] and L2TPv3 [RFC3931], which not only includes all the L2-specific specializations of L2TP, but also derivatives such as the Keyed IPv6 Tunnel [RFC8159]
- Generic Routing Encapsulation (GRE) [RFC2784] and Network Virtualization using GRE (NVGRE) [RFC7637]
- GPRS Tunnelling Protocol (GTP), specifically GTPv1 [GTPv1], GTP v1 User Plane [GTPv1-U], and GTP v2 Control Plane [GTPv2-C]

- Teredo [RFC4380]
- Control And Provisioning of Wireless Access Points (CAPWAP) [RFC5415]
- Locator/Identifier Separation Protocol (LISP) [RFC9300]
- Automatic Multicast Tunneling (AMT) [RFC7450]
- Virtual eXtensible Local Area Network (VXLAN) [RFC7348] and VXLAN-GPE [NVO3-VXLAN-GPE]
- The Network Service Header (NSH) [RFC8300] for Service Function Chaining (SFC)
- Geneve [RFC8926]
- Direct tunnelling of an IP packet within a UDP/IP datagram (see Section 3.1.11 of [RFC8085])
- TCP Encapsulation of Internet Key Exchange Protocol (IKE) and IPsec Packets (see Section 9.5 of [RFC9329])

Some of the listed protocols enable encapsulation of a variety of network layer protocols as inner and/or outer. This specification applies to the cases where there is an inner and outer IP header as described in Section 3. Otherwise, [RFC9599] gives guidance on how to design propagation of ECN into other protocols that might encapsulate IP.

Where protocols in the above list need to be updated to specify ECN propagation and are under IETF change control, update text is given in the following subsections. For those not under IETF control, it is **RECOMMENDED** that implementations of encapsulation and decapsulation comply with [RFC6040]. It is also **RECOMMENDED** that their specifications are updated to add a requirement to comply with [RFC6040] (as updated by the present document).

PPTP is not under the change control of the IETF, but it has been documented in an Informational RFC [RFC2637]. However, there is no need for the present specification to update PPTP because L2TP has been developed as a standardized replacement.

NVGRE is not under the change control of the IETF, but it has been documented in an Informational RFC [RFC7637]. NVGRE is a specific use case of GRE (it re-purposes the key field from the initial specification of GRE [RFC1701] as a Virtual Subnet ID). Therefore, the text that updates GRE in Section 6.1.2 below is also intended to update NVGRE.

Although the definition of the various GTP shim headers is under the control of the Third Generation Partnership Project (3GPP), it is hard to determine whether the 3GPP or the IETF controls standardization of the *process* of adding both a GTP and an IP header to an inner IP header. Nonetheless, the present specification is provided so that the 3GPP can refer to it from any of its own specifications of GTP and IP header processing.

The specification of CAPWAP already specifies [RFC3168] ECN propagation and ECN capability negotiation. Without modification, the CAPWAP specification already interworks with the backward-compatible updates to [RFC3168] in [RFC6040].

LISP made the ECN propagation procedures in [RFC3168] mandatory from the start. [RFC3168] has since been updated by [RFC6040], but the changes are backwards compatible, so there is still no need for LISP tunnel endpoints to negotiate their ECN capabilities.

VXLAN is not under the change control of the IETF, but it has been documented in an Informational RFC. In contrast, Generic Protocol Extension for VXLAN (VXLAN-GPE) is being documented under IETF change control. It is **RECOMMENDED** that VXLAN and VXLAN-GPE implementations comply with [RFC6040] when the VXLAN header is inserted between (or removed from between) IP headers. The authors of any future update to these specifications are encouraged to add a requirement to comply with [RFC6040] as updated by the present specification.

The Network Service Header (NSH) [RFC8300] has been defined as a shim-based encapsulation to identify the Service Function Path (SFP) in the Service Function Chaining (SFC) architecture [RFC7665]. A proposal has been made for the processing of ECN when handling transport encapsulation [RFC9600].

The specification of Geneve already refers to [RFC6040] for ECN encapsulation.

Section 3.1.11 of [RFC8085] already explains that a tunnel that encapsulates an IP header within a UDP/IP datagram needs to follow [RFC6040] when propagating the ECN field between inner and outer IP headers. Section 3 updates [RFC6040] to clarify that its scope includes cases with a shim header between the IP headers. It indirectly updates the scope of [RFC8085] to include cases with a shim header as well as a UDP header between the IP headers.

The requirements in Section 4 update [RFC6040], and hence indirectly update the UDP usage guidelines in [RFC8085] to add the important but previously unstated requirement that, if the UDP tunnel egress does not (or might not) support ECN propagation, a UDP tunnel ingress has to clear the outer IP ECN field to 0b00, e.g., by configuration.

Section 9.5 of [RFC9329] already recommends the compatibility mode of [RFC6040] in this case because there is not a one-to-one mapping between inner and outer packets.

6.1. Specific Updates to Protocols under IETF Change Control

6.1.1. L2TP (v2 and v3) ECN Extension

The L2TP terminology used here is defined in [RFC2661] and [RFC3931].

L2TPv3 [RFC3931] is used as a shim header between any packet-switched network (PSN) header (e.g., IPv4, IPv6, and MPLS) and many types of L2 headers. The L2TPv3 shim header encapsulates an L2-specific sub-layer, then an L2 header that is likely to contain an inner IP header (v4 or v6). Then this whole stack of headers can be encapsulated optionally within an outer UDP header then an outer PSN header that is typically IP (v4 or v6).

L2TPv2 is used as a shim header between any PSN header and a PPP header that is likely to encapsulate an IP header.

Even though these shims are rather fat (particularly in the case of L2TPv3), they still fit the definition of a tightly coupled shim header over an encapsulating header (Section 3.1) because all the headers encapsulating the L2 header are added (or removed) together. L2TPv2 and L2TPv3 are therefore within the scope of [RFC6040], as updated by Section 3.

Implementation of the ECN extension to L2TPv2 and L2TPv3 defined in Section 6.1.1.2 is **RECOMMENDED** in order to provide the benefits of ECN [RFC8087] whenever a node within an L2TP tunnel becomes the bottleneck for an end-to-end traffic flow.

6.1.1.1. Safe Configuration of a "Non-ECN" Ingress LCCE

The following text is appended to both Section 5.3 of [RFC2661] and Section 4.5 of [RFC3931] as an update to the base L2TPv2 and L2TPv3 specifications:

> The operator of an LCCE that does not support the ECN extension in Section 6.1.1.2 of RFC 9601 **MUST** follow the configuration requirements in Section 4 of RFC 9601 to ensure it clears the outer IP ECN field to 0b00 when the outer PSN header is IP (v4 or v6).

In particular, for an L2TP Control Connection Endpoint (LCCE) implementation that does not support the ECN extension, this means that configuration of how it propagates the ECN field between inner and outer IP headers **MUST** be independent of any configuration of the Diffserv extension of L2TP [RFC3308].

6.1.1.2. ECN Extension for L2TP (v2 or v3)

When the outer PSN header and the payload inside the L2 header are both IP (v4 or v6), an LCCE will follow the rules for propagation of the ECN field at ingress and egress in Section 4 of [RFC6040] to comply with [RFC6040].

Before encapsulating any data packets, [RFC6040] requires an ingress LCCE to check that the egress LCCE supports ECN propagation as defined in [RFC6040] or one of its compatible predecessors ([RFC4301] or the full functionality mode of [RFC3168]). If the egress supports ECN propagation, the ingress LCCE can use the normal mode of encapsulation (copying the ECN field from inner to outer). Otherwise, the ingress LCCE has to use compatibility mode [RFC6040] (clearing the outer IP ECN field to 0b00).

An LCCE can determine the remote LCCE's support for ECN either statically (by configuration) or by dynamic discovery during setup of each control connection between the LCCEs using the ECN Capability Attribute-Value Pair (AVP) defined in Section 6.1.1.2.1.

Where the outer PSN header is some protocol other than IP that supports ECN, the appropriate ECN propagation specification will need to be followed, e.g., [RFC5129]. Where no specification exists for ECN propagation by a particular PSN, [RFC9599] gives general guidance on how to design ECN propagation into a protocol that encapsulates IP.

6.1.1.2.1. ECN Capability AVP for Negotiation between LCCEs

The ECN Capability AVP defined here has Attribute Type 103. The AVP has the following format:

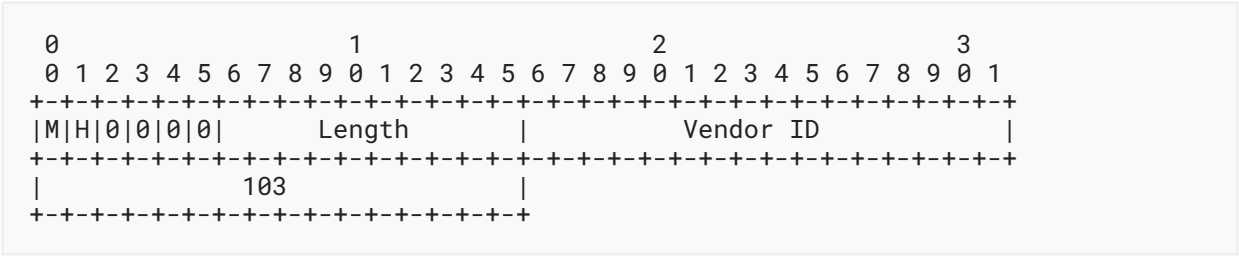

Figure 1: ECN Capability AVP for L2TP (v2 or v3)

This AVP **MAY** be present in the Start-Control-Connection-Request (SCCRQ) and Start-Control-Connection-Reply (SCCRP) message types. This AVP **MAY** be hidden (the H-bit is set to 0 or 1) and is optional (the M-bit is not set). The length (before hiding) of this AVP is 6 octets. The Vendor ID is the IETF Vendor ID of 0.

When an LCCE sends an ECN Capability AVP, it indicates that it supports ECN propagation. When no ECN Capability AVP is present, it indicates that the sender does not support ECN propagation.

If an LCCE initiating a control connection supports ECN propagation, it will send an SCCRQ containing an ECN Capability AVP. If the tunnel terminator supports ECN, it will return an SCCRP that also includes an ECN Capability AVP. Then, for any sessions created by that control connection, both ends of the tunnel can use the normal mode of [RFC6040]; i.e., they can copy the IP ECN field from inner to outer when encapsulating data packets.

On the other hand, if the tunnel terminator does not support ECN, it will ignore the ECN Capability AVP and send an SCCRP to the tunnel initiator without an ECN Capability AVP. The tunnel initiator interprets the absence of the ECN Capability flag in the SCCRP as an indication that the tunnel terminator is incapable of supporting ECN. When encapsulating data packets for any sessions created by that control connection, the tunnel initiator will then use the compatibility mode of [RFC6040] to clear the ECN field of the outer IP header to 0b00.

If the tunnel terminator does not support this ECN extension, the network operator is still expected to configure it to comply with the safety provisions set out in Section 6.1.1.1 when it acts as an ingress LCCE.

6.1.2. GRE

The GRE terminology used here is defined in [RFC2784]. GRE is often used as a tightly coupled shim header between IP headers. Sometimes, the GRE shim header encapsulates an L2 header, which might in turn encapsulate an IP header. Therefore, GRE is within the scope of [RFC6040] as updated by Section 3.

Implementation of support for [RFC6040] as updated by the present specification is **RECOMMENDED** for GRE tunnel endpoints in order to provide the benefits of ECN [RFC8087] whenever a node within a GRE tunnel becomes the bottleneck for an end-to-end IP traffic flow tunnelled over GRE using IP as the delivery protocol (outer header).

GRE itself does not support dynamic setup and configuration of tunnels. However, control plane protocols, such as Next Hop Resolution Protocol (NHRP) [RFC2332], Mobile IPv4 (MIP4) [RFC5944], Mobile IPv6 (MIP6) [RFC6275], Proxy Mobile IP (PMIP) [RFC5845], and IKEv2 [RFC7296], are sometimes used to set up GRE tunnels dynamically.

When these control protocols set up IP-in-IP or IPSec tunnels, it is likely that the resulting tunnels will propagate the ECN field as defined in [RFC6040] or one of its compatible predecessors ([RFC4301] or the full functionality mode of [RFC3168]). However, if they use a GRE encapsulation, this presumption is less sound.

Therefore, if the outer delivery protocol is IP (v4 or v6), the operator is obliged to follow the safe configuration requirements in Section 4. Section 6.1.2.1 updates the base GRE specification with this requirement to emphasize its importance.

Where the delivery protocol is some protocol other than IP that supports ECN, the appropriate ECN propagation specification will need to be followed, e.g., [RFC5129]. Where no specification exists for ECN propagation by a particular PSN, [RFC9599] gives more general guidance on how to propagate ECN to and from protocols that encapsulate IP.

6.1.2.1. Safe Configuration of a "Non-ECN" GRE Ingress

The following text is appended to Section 3 of [RFC2784] as an update to the base GRE specification:

> The operator of a GRE tunnel ingress **MUST** follow the configuration requirements in Section 4 of RFC 9601 when the outer delivery protocol is IP (v4 or v6).

6.1.3. Teredo

Teredo [RFC4380] provides a way to tunnel IPv6 over an IPv4 network with a UDP-based shim header between the two.

For Teredo tunnel endpoints to provide the benefits of ECN, the Teredo specification would have to be updated to include negotiation of the ECN capability between Teredo tunnel endpoints. Otherwise, it would be unsafe for a Teredo tunnel ingress to copy the ECN field to the IPv6 outer.

Those implementations known to the authors at the time of writing do not support propagation of ECN, but they do safely zero the ECN field in the outer IPv6 header. However, the specification does not mention anything about this.

To make existing Teredo deployments safe, it would be possible to add ECN capability negotiation to those that are subject to remote OS update. However, for those implementations not subject to remote OS update, it will not be feasible to require them to be configured correctly because Teredo tunnel endpoints are generally deployed on hosts.

Therefore, until ECN support is added to the specification of Teredo, the only feasible further safety precaution available here is to update the specification of Teredo implementations with the following text as a new section:

> 5.1.3. Safe "Non-ECN" Teredo Encapsulation
>
> A Teredo tunnel ingress implementation that does not support ECN propagation as defined in [RFC6040] or one of its compatible predecessors ([RFC4301] or the full functionality mode of [RFC3168]) **MUST** zero the ECN field in the outer IPv6 header.

6.1.4. AMT

AMT [RFC7450] is a tightly coupled shim header that encapsulates an IP packet and is encapsulated within a UDP/IP datagram. Therefore, AMT is within the scope of [RFC6040] as updated by Section 3.

Implementation of support for [RFC6040] as updated by the present specification is **RECOMMENDED** for AMT tunnel endpoints in order to provide the benefits of ECN [RFC8087] whenever a node within an AMT tunnel becomes the bottleneck for an IP traffic flow tunnelled over AMT.

To comply with [RFC6040], an AMT relay and gateway will follow the rules for propagation of the ECN field at ingress and egress, respectively, as described in Section 4 of [RFC6040].

Before encapsulating any data packets, [RFC6040] requires an ingress AMT relay to check that the egress AMT gateway supports ECN propagation as defined in [RFC6040] or one of its compatible predecessors ([RFC4301] or the full functionality mode of [RFC3168]). If the egress gateway supports ECN, the ingress relay can use the normal mode of encapsulation (copying the IP ECN field from inner to outer). Otherwise, the ingress relay has to use compatibility mode, which means it has to clear the outer ECN field to zero [RFC6040].

An AMT tunnel is created dynamically (not manually), so the relay will need to determine the remote gateway's support for ECN using the ECN capability declaration defined in Section 6.1.4.2.

6.1.4.1. Safe Configuration of a "Non-ECN" Ingress AMT Relay

The following text is appended to Section 4.2.2 of [RFC7450] as an update to the AMT specification:

> The operator of an AMT relay that does not support [RFC6040] or one of its compatible predecessors ([RFC4301] or the full functionality mode of [RFC3168]) **MUST** follow the configuration requirements in Section 4 of RFC 9601 to ensure it clears the outer IP ECN field to zero.

6.1.4.2. ECN Capability Declaration of an AMT Gateway

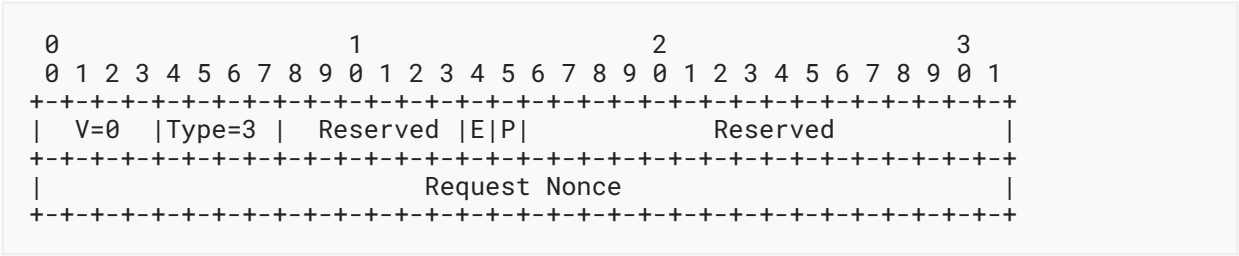

Figure 2: Updated AMT Request Message Format

Bit 14 of the AMT Request Message counting from 0 (or bit 7 of the Reserved field counting from 1) is defined here as the AMT Gateway ECN Capability flag (E) as shown in Figure 2. The definitions of all other fields in the AMT Request Message are unchanged from [RFC7450].

When the E flag is set to 1, it indicates that the sender of the message supports [RFC6040] ECN propagation. When it is cleared to zero, it indicates the sender of the message does not support [RFC6040] ECN propagation. An AMT gateway "that supports [RFC6040] ECN propagation" means one that propagates the ECN field to the forwarded data packet based on the combination of arriving inner and outer ECN fields as defined in Section 4 of [RFC6040].

The other bits of the Reserved field remain reserved. They will continue to be cleared to zero when sent and ignored when either received or forwarded as specified in Section 5.1.3.3 of [RFC7450].

An AMT gateway that does not support [RFC6040] **MUST NOT** set the E flag of its Request Message to 1.

An AMT gateway that supports [RFC6040] ECN propagation **MUST** set the E flag of its Relay Discovery Message to 1.

The action of the corresponding AMT relay that receives a Request message with the E flag set to 1 depends on whether the relay itself supports [RFC6040] ECN propagation:

- If the relay supports [RFC6040] ECN propagation, it will store the ECN capability of the gateway along with its address. Then, whenever it tunnels datagrams towards this gateway, it **MUST** use the normal mode of [RFC6040] to propagate the ECN field when encapsulating datagrams (i.e., it copies the IP ECN field from inner to outer).
- If the discovered AMT relay does not support [RFC6040] ECN propagation, it will ignore the E flag in the Reserved field as per Section 5.1.3.3 of [RFC7450].

  If the AMT relay does not support [RFC6040] ECN propagation, the network operator is still expected to configure it to comply with the safety provisions set out in Section 6.1.4.1.

7. IANA Considerations

IANA has assigned the following AVP in the L2TP "Control Message Attribute Value Pairs" registry:

Attribute Type	Description	Reference
103	ECN Capability	RFC 9601

Table 1

8. Security Considerations

The Security Considerations in [RFC6040] and [RFC9599] apply equally to the scope defined for the present specification.

9. References

9.1. 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>.

[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, DOI 10.17487/RFC2474, December 1998, <https://www.rfc-editor.org/info/rfc2474>.

[RFC2661] Townsley, W., Valencia, A., Rubens, A., Pall, G., Zorn, G., and B. Palter, "Layer Two Tunneling Protocol "L2TP"", RFC 2661, DOI 10.17487/RFC2661, August 1999, <https://www.rfc-editor.org/info/rfc2661>.

[RFC2784] Farinacci, D., Li, T., Hanks, S., Meyer, D., and P. Traina, "Generic Routing Encapsulation (GRE)", RFC 2784, DOI 10.17487/RFC2784, March 2000, <https://www.rfc-editor.org/info/rfc2784>.

[RFC3168] Ramakrishnan, K., Floyd, S., and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, DOI 10.17487/RFC3168, September 2001, <https://www.rfc-editor.org/info/rfc3168>.

[RFC3931] Lau, J., Ed., Townsley, M., Ed., and I. Goyret, Ed., "Layer Two Tunneling Protocol - Version 3 (L2TPv3)", RFC 3931, DOI 10.17487/RFC3931, March 2005, <https://www.rfc-editor.org/info/rfc3931>.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <https://www.rfc-editor.org/info/rfc4301>.

[RFC4380] Huitema, C., "Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)", RFC 4380, DOI 10.17487/RFC4380, February 2006, <https://www.rfc-editor.org/info/rfc4380>.

[RFC5129] Davie, B., Briscoe, B., and J. Tay, "Explicit Congestion Marking in MPLS", RFC 5129, DOI 10.17487/RFC5129, January 2008, <https://www.rfc-editor.org/info/rfc5129>.

[RFC6040] Briscoe, B., "Tunnelling of Explicit Congestion Notification", RFC 6040, DOI 10.17487/RFC6040, November 2010, <https://www.rfc-editor.org/info/rfc6040>.

[RFC6660] Briscoe, B., Moncaster, T., and M. Menth, "Encoding Three Pre-Congestion Notification (PCN) States in the IP Header Using a Single Diffserv Codepoint (DSCP)", RFC 6660, DOI 10.17487/RFC6660, July 2012, <https://www.rfc-editor.org/info/rfc6660>.

[RFC7450] Bumgardner, G., "Automatic Multicast Tunneling", RFC 7450, DOI 10.17487/RFC7450, February 2015, <https://www.rfc-editor.org/info/rfc7450>.

[RFC9599] Briscoe, B. and J. Kaippallimalil, "Guidelines for Adding Congestion Notification to Protocols that Encapsulate IP", RFC 9599, DOI 10.17487/RFC9599, June 2024, <https://www.rfc-editor.org/info/rfc9599>.

9.2. Informative References

[decap-test] Briscoe, B., "A Test for IP-ECN Propagation by a Remote Tunnel Endpoint", Technical Report, TR-BB-2023-003, DOI 10.48550/arXiv.2311.16825, November 2023, <https://arxiv.org/abs/2311.16825>.

[GTPv1] 3GPP, "General Packet Radio Service (GPRS); GPRS Tunnelling Protocol (GTP) across the Gn and Gp interface", Technical Specification 29.060.

[GTPv1-U] 3GPP, "General Packet Radio System (GPRS) Tunnelling Protocol User Plane (GTPv1-U)", Technical Specification 29.281.

[GTPv2-C] 3GPP, "3GPP Evolved Packet System (EPS); Evolved General Packet Radio Service (GPRS) Tunnelling Protocol for Control plane (GTPv2-C); Stage 3", Technical Specification 29.274.

[INTAREA-TUNNELS] Touch, J. D. and M. Townsley, "IP Tunnels in the Internet Architecture", Work in Progress, Internet-Draft, draft-ietf-intarea-tunnels-13, 26 March 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-intarea-tunnels-13>.

[NVO3-VXLAN-GPE] Maino, F., Kreeger, L., and U. Elzur, "Generic Protocol Extension for VXLAN (VXLAN-GPE)", Work in Progress, Internet-Draft, draft-ietf-nvo3-vxlan-gpe-13, 4 November 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-nvo3-vxlan-gpe-13>.

[RFC1701] Hanks, S., Li, T., Farinacci, D., and P. Traina, "Generic Routing Encapsulation (GRE)", RFC 1701, DOI 10.17487/RFC1701, October 1994, <https://www.rfc-editor.org/info/rfc1701>.

[RFC2332] Luciani, J., Katz, D., Piscitello, D., Cole, B., and N. Doraswamy, "NBMA Next Hop Resolution Protocol (NHRP)", RFC 2332, DOI 10.17487/RFC2332, April 1998, <https://www.rfc-editor.org/info/rfc2332>.

[RFC2637] Hamzeh, K., Pall, G., Verthein, W., Taarud, J., Little, W., and G. Zorn, "Point-to-Point Tunneling Protocol (PPTP)", RFC 2637, DOI 10.17487/RFC2637, July 1999, <https://www.rfc-editor.org/info/rfc2637>.

[RFC2983] Black, D., "Differentiated Services and Tunnels", RFC 2983, DOI 10.17487/RFC2983, October 2000, <https://www.rfc-editor.org/info/rfc2983>.

[RFC3260] Grossman, D., "New Terminology and Clarifications for Diffserv", RFC 3260, DOI 10.17487/RFC3260, April 2002, <https://www.rfc-editor.org/info/rfc3260>.

[RFC3308] Calhoun, P., Luo, W., McPherson, D., and K. Peirce, "Layer Two Tunneling Protocol (L2TP) Differentiated Services Extension", RFC 3308, DOI 10.17487/RFC3308, November 2002, <https://www.rfc-editor.org/info/rfc3308>.

[RFC5415] Calhoun, P., Ed., Montemurro, M., Ed., and D. Stanley, Ed., "Control And Provisioning of Wireless Access Points (CAPWAP) Protocol Specification", RFC 5415, DOI 10.17487/RFC5415, March 2009, <https://www.rfc-editor.org/info/rfc5415>.

[RFC5845] Muhanna, A., Khalil, M., Gundavelli, S., and K. Leung, "Generic Routing Encapsulation (GRE) Key Option for Proxy Mobile IPv6", RFC 5845, DOI 10.17487/RFC5845, June 2010, <https://www.rfc-editor.org/info/rfc5845>.

[RFC5944] Perkins, C., Ed., "IP Mobility Support for IPv4, Revised", RFC 5944, DOI 10.17487/RFC5944, November 2010, <https://www.rfc-editor.org/info/rfc5944>.

[RFC6275] Perkins, C., Ed., Johnson, D., and J. Arkko, "Mobility Support in IPv6", RFC 6275, DOI 10.17487/RFC6275, July 2011, <https://www.rfc-editor.org/info/rfc6275>.

[RFC7059] Steffann, S., van Beijnum, I., and R. van Rein, "A Comparison of IPv6-over-IPv4 Tunnel Mechanisms", RFC 7059, DOI 10.17487/RFC7059, November 2013, <https://www.rfc-editor.org/info/rfc7059>.

[RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T. Kivinen, "Internet Key Exchange Protocol Version 2 (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October 2014, <https://www.rfc-editor.org/info/rfc7296>.

[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, DOI 10.17487/RFC7348, August 2014, <https://www.rfc-editor.org/info/rfc7348>.

[RFC7637] Garg, P., Ed. and Y. Wang, Ed., "NVGRE: Network Virtualization Using Generic Routing Encapsulation", RFC 7637, DOI 10.17487/RFC7637, September 2015, <https://www.rfc-editor.org/info/rfc7637>.

[RFC7665] Halpern, J., Ed. and C. Pignataro, Ed., "Service Function Chaining (SFC) Architecture", RFC 7665, DOI 10.17487/RFC7665, October 2015, <https://www.rfc-editor.org/info/rfc7665>.

[RFC8085] Eggert, L., Fairhurst, G., and G. Shepherd, "UDP Usage Guidelines", BCP 145, RFC 8085, DOI 10.17487/RFC8085, March 2017, <https://www.rfc-editor.org/info/rfc8085>.

[RFC8087] Fairhurst, G. and M. Welzl, "The Benefits of Using Explicit Congestion Notification (ECN)", RFC 8087, DOI 10.17487/RFC8087, March 2017, <https://www.rfc-editor.org/info/rfc8087>.

[RFC8159] Konstantynowicz, M., Ed., Heron, G., Ed., Schatzmayr, R., and W. Henderickx, "Keyed IPv6 Tunnel", RFC 8159, DOI 10.17487/RFC8159, May 2017, <https://www.rfc-editor.org/info/rfc8159>.

[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>.

[RFC8300] Quinn, P., Ed., Elzur, U., Ed., and C. Pignataro, Ed., "Network Service Header (NSH)", RFC 8300, DOI 10.17487/RFC8300, January 2018, <https://www.rfc-editor.org/info/rfc8300>.

[RFC8311] Black, D., "Relaxing Restrictions on Explicit Congestion Notification (ECN) Experimentation", RFC 8311, DOI 10.17487/RFC8311, January 2018, <https://www.rfc-editor.org/info/rfc8311>.

[RFC8926] Gross, J., Ed., Ganga, I., Ed., and T. Sridhar, Ed., "Geneve: Generic Network Virtualization Encapsulation", RFC 8926, DOI 10.17487/RFC8926, November 2020, <https://www.rfc-editor.org/info/rfc8926>.

[RFC9300] Farinacci, D., Fuller, V., Meyer, D., Lewis, D., and A. Cabellos, Ed., "The Locator/ID Separation Protocol (LISP)", RFC 9300, DOI 10.17487/RFC9300, October 2022, <https://www.rfc-editor.org/info/rfc9300>.

[RFC9329] Pauly, T. and V. Smyslov, "TCP Encapsulation of Internet Key Exchange Protocol (IKE) and IPsec Packets", RFC 9329, DOI 10.17487/RFC9329, November 2022, <https://www.rfc-editor.org/info/rfc9329>.

[RFC9331] De Schepper, K. and B. Briscoe, Ed., "The Explicit Congestion Notification (ECN) Protocol for Low Latency, Low Loss, and Scalable Throughput (L4S)", RFC 9331, DOI 10.17487/RFC9331, January 2023, <https://www.rfc-editor.org/info/rfc9331>.

[RFC9600] Eastlake, D., Briscoe, B., Zhuang, S., Malis, A., and X. Wei, "Explicit Congestion Notification (ECN) and Congestion Feedback Using the Network Service Header (NSH) and IPFIX", RFC 9600, DOI 10.17487/RFC9600, June 2024, <https://www.rfc-editor.org/info/rfc9600>.

Acknowledgements

Thanks to Ing-jyh (Inton) Tsang for initial discussions on the need for ECN propagation in L2TP and its applicability. Thanks also to Carlos Pignataro, Tom Herbert, Ignacio Goyret, Alia Atlas, Praveen Balasubramanian, Joe Touch, Mohamed Boucadair, David Black, Jake Holland, Sri Gundavelli, Gorry Fairhurst, and Martin Duke for helpful advice and comments. [RFC7059] helped to identify a number of tunnelling protocols to include within the scope of this document.

Bob Briscoe Bob Briscoe was part-funded by the Research Council of Norway through the TimeIn project for early drafts, and he was funded by Apple Inc. for later draft versions (from -17). The views expressed here are solely those of the authors.

Author's Address

Bob Briscoe
Independent
United Kingdom
Email: ietf@bobbriscoe.net
URI: https://bobbriscoe.net/