IPsec Interactions with ECN

DRAFT - February 24 - NOT YET SUBMITTED

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

Abstract

IPsec supports secure communication over potentially insecure network components such as intermediate routers. IPsec protocols support two operating modes, transport mode and tunnel mode. Explicit Congestion Notification (ECN) is an experimental addition to the IP architecture that provides indication of onset of congestion to delay- or loss-sensitive applications. ECN provides the congestion indication so as to enable adaptation to network conditions without the impact of dropped packets [RFC 2481]. Currently, the way ECN is specified does not conform to the manner in which IPsec tunnels are defined to be used. This document considers issues related to interactions between ECN and IPsec tunnel mode, and proposes two alternative solutions.

1. Introduction.

IPsec supports secure communication over potentially insecure network components such as intermediate routers. IPsec protocols support two operating modes, transport mode and tunnel mode that span a wide range of security requirements and operating environments. Transport mode security protocol header(s) are inserted between the IP (IPv4 or IPv6) header and higher layer protocol headers (e.g., TCP), and hence transport mode can only be used for end-to-end security on a connection. IPsec tunnel mode is based on adding a new "outer" IP header that encapsulates the original, or "inner" IP header and its associated packet. Tunnel mode security headers are inserted between these two IP headers. In contrast to transport mode, the new "outer" IP header and tunnel mode security headers may be added and removed at intermediate points along a connection, enabling security gateways to secure vulnerable portions of a connection without requiring endpoint participation in the security protocols. An important aspect of tunnel mode security is that the outer header is discarded at tunnel egress, ensuring that security threats based on modifying the IP header do not propagate beyond that tunnel endpoint. Further discussion of IPsec can be found in [RFC 2401].

Explicit Congestion Notification (ECN) is an experimental addition to the IP architecture that provides congestion indication to delay- or loss-sensitive applications to enable them to adapt to network conditions without the impact of dropped packets [RFC 2481]. An ECN-capable router uses the ECN mechanism to signal congestion to connection endpoints by setting a bit in the IP header. These endpoints then react as if a packet had been dropped (e.g., TCP halves its congestion window). This ability to avoid dropping packets in response to congestion is supported by the use of active queue management mechanisms (e.g., RED) in routers; such mechanisms begin to mark or drop packets as a consequence of congestion before a congested router queue is completely full. ECN is defined to be used as an optimization -- routers are not required to support ECN, and even an ECN-capable router may drop packets from ECN-capable connections when necessary. The advantage to a router of not dropping such packets is that ECN can provide a more timely reaction to congestion than reactions based on drop detection via duplicate ACKs or timeout.

Currently, the way ECN is specified does not conform to the manner in which IPsec tunnels are defined to be used. Current use of ECN over an IPsec tunnel results in routers attempting to use the outer IP header to signal congestion to endpoints, but those congestion warnings never arrive because the outer header is discarded. RFC 2481 currently recommends that ECN not be used with IPsec tunnels in order to avoid this behavior and its consequences.

This document considers issues related to interactions between ECN and IPsec tunnel mode. In principle the use of ECN in the outer header of an IPsec tunnel raises security concerns because an adversary could tamper with the ECN information that propagates beyond the tunnel endpoint. Based on an analysis (included in this document) of these concerns and the resultant risks, our overall approach is to make support for ECN a configurable part of an IPsec tunnel mode Security Association (SA), so that ECN can be disabled for an IPsec tunnel in situations where the risks of using ECN are judged to outweigh its benefits. The result is that the security administrator is presented with two options for the behavior of ECN-capable connections over an IPsec tunnel:

- A limited-functionality option in which ECN is preserved in the inner header, but disabled in the outer header. The only mechanism available for signaling congestion occurring within the tunnel in this case is dropped packets.
- A full functionality option that supports ECN in both the inner and outer headers, and propagates congestion warnings from nodes within the tunnel to endpoints.

Support for these options requires changes to IP header processing at tunnel ingress and egress. A subset of these changes sufficient to support only the limited-functionality option SHOULD be applied to all IPsec implementations in order to eliminate the current incompatibility between ECN and IPsec tunnels.

The main goal of this document is to give guidance about the tradeoffs between the limited-functionality and full-functionality options. This includes a full discussion of the potential effects of an adversary's modifications of the CE and ECT bits.

2. Architecture.

ECN uses two bits in the IP header (ECT - ECN Capable Transport, CE - Congestion Experienced) for signaling between routers and connection endpoints, and uses two flags in the TCP header (ECN-Echo - Echo ECN bit in IP header, CWR - Congestion Window Reduced) for TCP-endpoint to TCP-endpoint signaling. For a TCP connection, a typical sequence of events in an ECN-based reaction to congestion is as follows:

- The ECT bit is set in all packets transmitted by the sender to indicate that ECN is supported on this TCP connection.
- An ECN-capable router detects impending congestion and detects that the ECT bit is set in the packet it is about to drop. Instead of dropping the packet, the router sets the CE bit and forwards the packet.
- The receiver receives the packet with CE set, and sets the ECN-Echo flag in its next TCP ACK sent to the sender.
- The sender receives the TCP ACK with ECN-Echo set, and reacts to

the congestion as if a packet had been dropped.
- The sender sets the CWR flag in the TCP header of the next packet sent to the receiver to acknowledge its receipt of and reaction to the ECN-Echo flag.

Further details on ECN functionality including negotiation of ECN-capability as part of connection setup as well as the responsibilities and requirements of ECN-capable routers and transports can be found in [RFC2481].

ECN interacts with IPsec tunnels because the two ECN bits in the IP header are in what IPsec refers to as the IPv4 TOS octet or IPv6 Traffic Class octet; this field is copied or mapped from the inner IP header to the outer IP header at IPsec tunnel ingress, and the outer header's copy of this field is discarded at IPsec tunnel egress [RFC2401]. If an ECN-capable router were to set the CE (Congestion Experienced) bit in an IPsec-tunneled packet, this indication would be discarded at tunnel egress, losing the indication of congestion. As a consequence of this behavior, ECN usage over IPsec tunnels is not currently recommended [RFC2481].

The IPsec limited-functionality option for ECN encapsulation is for the ECT bit in the outside (encapsulating) header to be off (i.e., set to 0), regardless of the value of the ECT bit in the inside (encapsulated) header. With this option, the ECN field in the inner header is not altered upon de-capsulation. The disadvantage of this approach is that the flow does not have ECN support for that part of the path that is using IPsec tunneling, even if the encapsulated packet is ECN-Capable. That is, if the encapsulated packet arrives at a congested router that is ECN-capable, and the router decides to drop or mark the packet as an indication of congestion to the end nodes, the router will not be permitted to set the CE bit in the packet header, but instead will have to drop the packet.

The IPsec full-functionality option for ECN encapsulation follows the description in Section 10.1 of RFC 2481 of tunneling with ECN. This option is to copy the ECT bit of the inside header to the outside header on encapsulation, and to OR the CE bit from the outer header with the CE bit of the inside header on decapsulation. With the full-functionality option, a flow can take advantage of ECN for those parts of the path that might use IPsec tunneling. The disadvantage of the full-functionality option is that IPsec cannot protect the flow from certain modifications to the ECN bits in the IP header within the tunnel. The potential dangers from modifications to the ECN bits in the IP header are described in detail in Section 4 below.

This document proposes minor changes to IPsec in order to enable ECN experimentation over IPsec tunnels, and avoid losing congestion

indications in the case that an ECN-capable router or routers are traversed by an IPsec tunnel carrying ECN-capable connections. In summary, two changes are proposed to IPsec functionality:

(1) Modify the handling of the IPv4 TOS octet and IPv6 Traffic Class octet at IPsec tunnel endpoints to prevent loss of congestion indications when an IPsec tunnel traverses an ECN-capable router.

(2) Enable the endpoints of an IPsec tunnel to negotiate the usage of ECN in the outer headers of that tunnel based on security policy. ECN is only used in the outer header of packets from connections that support ECN.

The minimum required to make ECN usable with IPsec tunnels is a simplified version of the first change that prevents ECN from being enabled in the outer header of an IPsec tunnel. Full support for ECN requires the ability to negotiate ECN usage between tunnel endpoints, including the ability of a security administrator to disable ECN in situations where she believes the risks (e.g., of lost congestion notifications) outweigh the benefits of ECN.

3. IPsec Changes for ECN support

This section describes the detailed changes for support of ECN over IPsec tunnels, including the negotiation of ECN support between tunnel endpoints. In order to avoid the loss of congestion indication at tunnel egress, full ECN functionality for an IPsec tunnel requires that both ends of the tunnel agree that ECN is being used. This is supported by three changes to IPsec:

- A Security Association Database (SAD) field indicating whether tunnel encapsulation and decapsulation processing should allow or forbid ECN usage in the outer IP header.
- A Security Association Attribute for negotiation of this SAD field between the two endpoints of an SA that supports tunnel mode.
- Changes to tunnel mode encapsulation and decapsulation processing to allow or forbid ECN usage in the outer IP header based on the value of the SAD field. When ECN usage is allowed in the outer IP header, ECT is set in the outer header for ECN-capable connections and congestion indications (CE bit) from such connections are propagated to the inner header at tunnel egress.

The first two changes are OPTIONAL. The encapsulation and decapsulation processing changes are RECOMMENDED, but MAY be implemented without the other two changes by assuming that ECN usage is always forbidden. These changes are covered in the following three subsections.

3.1.1. ECN Tunnel Security Association Database Field

Full ECN functionality requires a new field be added to the SAD (see [RFC2401]):

> ECN Tunnel: allowed or forbidden.
> Indicates whether ECN-capable connections using this SA in tunnel mode may receive ECN congestion notifications for congestion occurring within the tunnel. The allowed value enables ECN congestion notifications. The forbidden value disables such notifications, causing all congestion to be indicated via dropped packets.
>
> [OPTIONAL. The value of this field SHOULD be assumed to be forbidden in implementations that do not support it.]

Support for this attribute REQUIRES that the SA specification in a Security Policy Database (SPD) entry support a corresponding attribute, and that this SPD attribute be covered by the SPD administrative interface described in Section 4.4.1 of [RFC2401].

3.1.2. ECN Tunnel Security Association Attribute

A new IPsec Security Association Attribute is defined to enable the use of ECN for IPsec tunnels to be negotiated (see [RFC2407]). This attribute is OPTIONAL, although any implementation that supports it SHOULD also support the SAD field defined in Section 3.1.1.

Attribute Type

class	value	type
ECN Tunnel	TBD	Basic

NB: The attribute identification value will be determined by IANA.

Class Values

ECN Tunnel

> Specifies whether ECN may be used with Tunnel Encapsulation Mode. This affects tunnel encapsulation and decapsulation processing - see Section 3.1.3.
>
> | | |
> |---|---|
> | RESERVED | 0 |
> | Allowed | 1 |
> | Forbidden | 2 |

Values 3-61439 are reserved to IANA. Values 61440-65535 are for private use.

If unspecified, the default shall be assumed to be Forbidden.

3.1.3. Changes to IPsec Tunnel Header Processing

Subsequent to the publication of the IPsec RFCs, the TOS octet of IPv4 and the Traffic Class octet of IPv6 have been superseded by the DS Field octet as defined in [RFC2474]. The two ECN bits in the IP header, ECT and CE, occupy bits 6 and 7 of the DS Field octet [RFC2481]. For full ECN support the encapsulation and decapsulation processing for the IPv4 TOS field and the IPv6 Traffic Class field are changed from what is specified in [RFC2401] to the following:

	<-- How Outer Hdr Relates to Inner Hdr --> Outer Hdr at Encapsulator	Inner Hdr at Decapsulator
IPv4 Header fields: DS Field	constructed (5)	constructed (7)
IPv6 Header fields: DS Field	constructed (6)	constructed (7)

(5)(6) Copy the Differentiated Services Codepoint (DSCP, bits 0-5). If the value of the ECN Tunnel field in the SAD entry for this SA is "allowed" and the value of ECT (bit 6) is 1 in the inner header, set ECT to 1 in the outer header, else set ECT to 0 in the outer header. Set CE (bit 7) to 0 in the outer header.

(7) If the value of the ECN tunnel field in the SAD entry for this SA is "allowed" and the value of ECT (bit 6) in the inner header is 1, and the value of CE (bit 7) in the outer header is 1, then set CE to 1 in the inner header, else make no change to this field of the inner header.

NB: (5) and (6) are identical to match usage in [RFC2401] where they are different.

This description applies to implementations that support the ECN Tunnel field in the SAD; such implementations MUST implement this processing of the DS field instead of the processing of the IPv4 TOS octet and IPv6 Traffic Class octet defined in [RFC2401]. This constitutes complete ECN support for IPsec tunnels.

An implementation that does not support the ECN Tunnel field in the SAD SHOULD implement processing of the DS Field by assuming that the

value of the ECN Tunnel field of the SAD is "forbidden" for every SA. In this case, the RECOMMENDED processing reduces to:

(5)(6) Copy the Differentiated Services Codepoint (DSCP, bits 0-5). Set bits 6 and 7 (ECT and CE) of the DS field to zero.
(7) Make no change to DS field in the inner header.

This constitutes partial ECN support for IPsec tunnels.

In addition, it is RECOMMENDED for that packets with ECN and CE both set to 1 in the outer header be dropped if they arrive on an SA that forbids or is assumed to forbid ECN usage in tunnel mode. This applies to both the complete ECN support and partial ECN support implementation approaches. This is motivated by backwards compatibility and is discussed further in Section 6.

4. Possible Changes to the ECN Field

This section considers the issues when a router is operating, possibly maliciously, to modify either of the bits in the ECN field. In this section we represent the ECN field in the IP header by the tuple (ECT bit, CE bit). The ECT bit, when set to 1, indicates an ECN-Capable Transport. The CE bit, when set to 1, indicates that Congestion was Experienced in the path.

By tampering with the bits in the ECN field, an adversary (or a broken router) could do one or more of the following: erase the ECN congestion indication, falsely report congestion, disable ECN-Capability for an individual packet, or falsely indicate ECN-Capability. We systematically examine the various cases by which the ECN field could be modified. The important criterion we consider in determining the consequences of such modifications is whether it is likely to lead to poorer behavior in any dimension (throughput, delay, fairness or functionality) than if a router were to drop a packet.

4.1. Erasing the Congestion Indication

First, we consider the changes that a router could make that would result in effectively erasing the congestion indication after it had been set by a router upstream. The convention followed is: (ECT, CE) of received packet -> (ECT, CE) of packet transmitted.

(1, 1) -> (1, 0): erase only the CE bit that was set.
(1, 1) -> (0, 0): erase both the ECT bit and the CE bit.
(1, 1) -> (0, 1): erase the ECT bit

The first change turns off the CE bit after it has been set by some

upstream router along the path. The consequence for the upstream router is that there is a potential for congestion to build for a time, because the congestion indication does not reach the source. However, the packet would be received and acknowledged.

The potential effect of erasing the congestion indication is complex, and is discussed in depth in Section 5 below. Note that the effect of erasing the congestion indication is different from dropping a packet in the network. When a data packet is dropped, the drop is detected by the TCP sender, and interpreted as an indication of congestion. Similarly, if a sufficient number of consecutive acknowledgement packets are dropped, causing the cumulative acknowledgement field not to be advanced at the sender, the sender is limited by the congestion window from sending additional packets, and ultimately the retransmit timer expires.

In contrast, a systematic erasure of the CE bit by a downstream router can have the effect of causing a queue buildup at an upstream router, including the possible loss of packets due to buffer overflow. There is a potential of unfairness in that another flow that goes through the congested router may react to the CE bit set while the flow that has the CE bit erased may see better performance. The limitations on this potential unfairness are discussed in more detail in Section 5 below.

The second change is to turn off both the ECT and the CE bits, thus erasing the congestion indication and disabling ECN-Capability at the same time. The third change turns off only the ECT bit, disabling ECN-Capability. The proposal in this Internet Draft is for the receiver at the end of a tunnel to copy the CE bit, if set, from the outer header to the inner header during decapsulation, if the ECT bit in the inner header is set and the tunnel provides full ECN support. In this case, the third change within an IPsec tunnel would not erase the congestion indication, but would only disable ECN-Capability for that packet within the rest of the tunnel. However, when performed outside of an IPsec tunnel, the third change would also effectively erase the congestion indication, because, from RFC 2481, an ECN field of (0, 1) is undefined.

The ‘erasure’ of the congestion indication is only effective if the packet does not end up being marked or dropped again by a downstream router. With the first change, the packet remains ECN-Capable, and could be either marked or dropped by a downstream router as an indication of congestion. With the second and third changes, the packet is no longer ECN-capable, and can therefore be dropped but not marked by a downstream router as an indication of congestion.

4.2. Falsely Reporting Congestion

(1, 0) -> (1, 1)

This change is to set the CE bit when the ECT bit was already set, even though there was no congestion. This change does not affect the treatment of that packet along the rest of the path. In particular, a router does not examine the CE bit in deciding whether to drop or mark an arriving packet.

However, this could result in the application unnecessarily invoking end-to-end congestion control, and reducing its arrival rate. By itself, this is no worse (for the application or for the network) than if the tampering router had actually dropped the packet.

4.3. Disabling ECN-Capability

(1, 0) -> (0, *)

This change is to turn off the ECT bit of a packet that does not have the CE bit set. (Section 4.1 discussed the case of turning off the ECT bit of a packet that does have the CE bit set.) This means that if the packet later encounters congestion (e.g., by arriving to a RED queue with a moderate average queue size), it will be dropped instead of being marked. By itself, this is no worse (for the application) than if the tampering router had actually dropped the packet. The saving grace in this particular case is that there is no congested router upstream expecting a reaction from setting the CE bit.

4.4. Falsely Indicating ECN-Capability

This change is to incorrectly label a packet as ECN-Capable.

(0, *) -> (1, 0);
(0, *) -> (1, 1);

If the packet later encounters moderate congestion at an ECN-Capable router, the router could set the CE bit instead of dropping the packet. If the transport protocol in fact is not ECN-Capable, then the transport will never receive this indication of congestion, and will not reduce its sending rate in response. The potential consequences of falsely indicating ECN-capability are discussed further in Section 5 below.

If the packet never later encounters congestion at an ECN-Capable router, then the first of these two changes would have no effect. The second change, however, would have the effect of giving false reports of congestion to a monitoring device along the path. If the

transport protocol is ECN-Capable, then the second of these two changes (when, for example, (0,0) was changed to (1,1)) could also have an effect at the transport level, by combining falsely indicating ECN-Capability with falsely reporting congestion. For an ECN-capable transport, this would cause the transport to unnecessarily react to congestion. In this particular case, the router that is incorrectly changing the ECN field could have dropped the packet. Thus for this case of an ECN-capable transport, the consequence of this change to the ECN field is no worse than dropping the packet.

4.5. Changes with No Functional Effect

(0, *) -> (0, *)

The CE bit is ignored in a packet that does not have the ECT bit set. Thus, this change would have no effect, in terms of ECN.

4.6. Information carried in the Transport Header

For TCP, an ECN-capable TCP receiver informs its TCP peer that it is ECN-capable at the TCP level, using information in the TCP header at the time the connection is setup. This document does not consider potential dangers introduced by changes in the transport header because the IPsec tunnel protects the transport header.

5. Implications of Subverting End-to-End Congestion Control

This section focuses on the potential repercussions of subverting end-to-end congestion control by either falsely indicating ECN-Capability, or by erasing the congestion indication in ECN (the CE-bit). Subverting end-to-end congestion control by either of these two methods can have consequences both for the application and for the network. We discuss these separately below.

The first method to subvert end-to-end congestion control, falsely indicating ECN-Capability, effectively subverts end-to-end congestion control only if the packet later encounters congestion that results in the setting of the CE bit. In this case, the transport protocol does not receive the indication of congestion from these downstream congested routers.

The second method to subvert end-to-end congestion control, ‘erasing’ the (set) CE bit in a packet, effectively subverts end-to-end congestion control only when the CE bit in the packet was set earlier by a congested router. In this case, the transport protocol does not receive the indication of congestion from the upstream congested routers.

Either of these two methods of subverting end-to-end congestion control can potentially introduce more damage to the network (and possibly to the flow itself) than if the adversary had simply dropped packets from that flow. However, as we discuss later in this section and in Section 7, this potential damage is limited.

5.1. Implications for the Network and for Competing Flows

The CE bit of the ECN field is only used by routers as an indication of congestion during periods of *moderate* congestion. From RFC 2481: "When severe congestion has occurred and the router's queue is full, then the router has no choice but to drop some packet when a new packet arrives." Although it is not explicitly mandated by RFC 2481, the general guidelines are that a router should drop rather than mark packets during heavy congestion even if the router's queue is not yet full. For example, for routers using active queue management based on RED, the router should drop rather than mark packets that arrive while the average queue sizes exceed the RED queue's maximum threshold.

One consequence for the network of subverting end-to-end congestion control is that flows that do not receive the congestion indications from the network might increase their sending rate until they drive the network into heavier congestion. Then, the congested router could begin to drop rather than mark arriving packets. For flows that are not isolated by some form of per-flow scheduling or other per-flow mechanisms, but that are instead aggregated with other flows in a single queue in an undifferentiated fashion, this packet-dropping at the congested router would apply to all flows that share that queue. Thus, the consequences would be to increase the level of congestion in the network.

In some cases, the increase in the level of congestion will lead to a substantial buffer buildup at the congested queue that will be sufficient to drive the congested queue from the packet-marking to the packet-dropping regime. This transition could occur either because of buffer overflow, or because of the active queue management