Internet Engineering Task Force
Floyd, Black, and Ramakrishnan

INTERNET DRAFT
draft-ipsec-ecn-00.txt February 1999

IPsec Interactions with ECN

DRAFT - February 24 - NOT YET SUBMITTED

Status of this Memo

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                 Inner Hdr at
   IPv4                 Encapsulator                 Decapsulator
     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 policy described above that drops packets when the average queue is above RED's maximum threshold. At this point, all flows, including the subverted flow, will begin to see packet drops instead of packet marks, and a malicious or broken router will no longer be able to 'erase' these indications of congestion in the network. If the end nodes are deploying appropriate end-to-end congestion control, then the subverted flow will reduce its arrival rate in response to congestion. When the level of congestion is sufficiently reduced, the congested queue can return from the packet-dropping regime to the packet-marking regime. The steady-state pattern could be one of the congested queue oscillating between these two regimes.

In other cases, the consequences of subverting end-to-end congestion control will not be severe enough to drive the congested link into sufficiently-heavy congestion that packets are dropped instead of being marked. In this case, the implications for competing flows in the network will be a slightly-increased rate of packet marking or dropping, and a corresponding decrease in the bandwidth available to those flows. This can be a stable state if the arrival rate of the subverted flow is sufficiently small, relative to the link bandwidth, that the average queue size at the congested router remains under control. In particular, the subverted flow could have a limited bandwidth demand on the link at this router, while still getting more than its "fair" share of the link. This limited demand could be due to a limited demand from the data source; a limitation from the TCP advertised window; a lower-bandwidth access pipe; or other factors. Thus the subversion of ECN-based congestion control can still lead to unfairness, which we believe is appropriate to note here.

The threat to the network posed by the subversion of ECN-based congestion control in the network is essentially the same as the threat posed by an end-system that intentionally fails to cooperate with end-to-end congestion control. The deployment of mechanisms in routers to address this threat is an open research question, and is discussed further in Section 7.

Let us take the example described in Section 4.1, where the CE bit that was set in a packet is erased by the tunnel: {(1, 1) -> (1, 0)}. The consequence for the congested upstream router that set the CE bit is that this congestion indication does not reach the end nodes for that flow. The source (even one which is completely cooperative and not malicious) is thus allowed to continue to increase its sending rate (if it is a TCP flow, by increasing its congestion window). The flow potentially achieves better throughput than the other flows that also share the congested router, especially if there are no policing mechanisms or per-flow queueing mechanisms at that router. Consider the behavior of the other flows, especially if they are cooperative: that is, the flows that do not experience subverted end-to-end congestion control. They are likely to reduce their load (e.g., by reducing their window size) on the congested router, thus benefiting our subverted flow. This results in unfairness. As we discussed above, this unfairness could either be transient (because the congested queue is driven into the packet-marking regime), oscillatory (because the congested queue oscillated between the packet marking and the packet dropping regime), or more moderate but a persistent stable state (because the congested queue is never driven to the packet dropping regime).

The results would be similar if the subverted flow was intentionally avoiding end-to-end congestion control. One difference is that a flow that is intentionally avoiding end-to-end congestion control at the end nodes can avoid end-to-end congestion control even when the congested queue is in packet-dropping mode, by refusing to reduce its sending rate in response to packet drops in the network. Thus the problems for the network of the subversion of ECN-based congestion control are less severe than the problems caused by the intentional avoidance of end-to-end congestion control in the end nodes. It is also the case that it is considerably more difficult to control the behavior of the end nodes than it is to control the behavior of the infrastructure itself. This is not to say that the problems for the network posed by the network's subversion of ECN-based congestion control are small; just that they are dwarfed by the problems for the network posed by the subversion of either ECN-based or packet-based congestion control by the end nodes.

5.2. Implications for the Subverted Flow

When a source indicates that it is ECN-capable, there is an expectation that the routers in the network that are capable of participating in ECN will use the CE bit for indication of congestion. There is the potential benefit of using ECN in reducing the amount of packet loss (in addition to the reduced queueing delays because of active queue management policies). When the packet flows through a tunnel where the nodes that the tunneled packets traverse are untrusted in some way, the expectation is that IPsec will protect the flow from subversion that results in undesirable consequences.

In many cases, a subverted flow will benefit from the subversion of end-to-end congestion control for that flow in the network, by receiving more bandwidth that it would have otherwise, relative to competing non-subverted flows. If the congested queue reaches the packet-dropping stage, then the subversion of end-to-end congestion control might or might not be of overall benefit to the subverted flow, depending on that flow's relative tradeoffs between throughput, loss, and delay.

One form of subverting end-to-end congestion control is to falsely indicate ECN-capability by setting the ECT bit. This has the consequence of downstream congested routers setting the CE bit in vain. However, as we describe in the section below, if the ECT bit is changed in the IPsec tunnel, this can be detected at the egress point of the tunnel.

The second form of subverting end-to-end congestion control is to erase the congestion indication, either by erasing the CE bit directly, or by erasing the ECT bit when the CE bit is already set. In this case, it is the upstream congested routers that set the CE bit in vain. There are several possible scenarios for this subversion of end-to-end congestion control within an IPsec tunnel. If the ECT bit is erased within an IPsec tunnel, then this can be detected at the egress point of the tunnel. If the CE bit is set upstream of the IPsec tunnel, then any erasure of the outer header's CE bit within the tunnel will have no effect because the inner header preserves the set value of the CE bit. However, if the CE bit is set within the tunnel, and erased either within or downstream of the tunnel, this is not necessarily detected at the egress point of the tunnel.

With this subversion of end-to-end congestion control, an end-system transport does not respond to the congestion indication. Along with the increased unfairness for the non-subverted flows described in the previous section, the congested router's queue could continue to build, resulting in packet loss at the congested router - which is a means for indicating congestion to the transport in any case. In the interim, the flow might experience higher queueing delays, possibly along with an increased bandwidth relative to other non-subverted flows. But transports do not inherently make assumptions of consistently experiencing carefully managed queueing in the path. We believe that these forms of subverting end-to-end congestion control are no worse for the subverted flow than if the adversary had simply dropped the packets of that flow itself.

5.3. Non-ECN-Based Methods of Subverting End-to-end Congestion Control

We have shown that, in many cases, a malicious or broken router that is able to change the bits in the ECN field can do no more damage than if it had simply dropped the packet in question. However, this is not true in all cases, in particular in the cases where the broken router subverted end-to-end congestion control by either falsely indicating ECN-Capability or by erasing the ECN congestion indication (in the CE-bit). While there are many ways that a router can harm a flow by dropping packets, a router cannot subvert end-to-end congestion control by dropping packets. As an example, a router cannot subvert TCP congestion control by dropping data packets, acknowledgement packets, or control packets.

Even though packet-dropping cannot be used to subvert end-to-end congestion control, there *are* non-ECN-based methods for subverting end-to-end congestion control that a broken or malicious router could use. For example, a broken router could duplicate data packets, thus effectively negating the effects of end-to-end congestion control along some portion of the path. (For a router that duplicated packets within an IPsec tunnel, the security administrator can cause the duplicate packets to be discarded by configuring anti-replay protection for the tunnel.) This duplication of packets within the network would have similar implications for the network and for the subverted flow as those described in Sections 5.1 and 5.2 above.

6. Changes to the ECN Field within an IPsec Tunnel.

The presence of a copy of the ECN field in the inner header of an IPsec tunnel mode packet provides an opportunity for detection of modifications to the ECT bit in the outer header. Comparison of the ECT bits in the inner and outer headers falls into two categories for implementations that conform to this document:

(a) If the SA allows ECN usage within the tunnel, then the values of the ECT bits in the inner and outer headers should be identical.

(b) If the SA disallows ECN usage within the tunnel, then the ECT bit in the outer header should be 0.

Receipt of a packet not satisfying the appropriate condition for its SA is an auditable event, but an implementation MAY create audit records with per-SA counts of incorrect packets over some time period rather than creating an audit record for each erroneous packet. Any such audit record SHOULD contain the headers from at least one erroneous packet, but need not contain the headers from every packet represented by the entry.

An important and likely situation involves an IPsec implementation not updated to this document’s requirements serving as tunnel ingress for a tunnel egress at an implementation that has been updated. The ECN Tunnel attribute cannot be negotiated in this case because the tunnel ingress implementation does not support it. If packets from an ECN-capable connection use this tunnel, ECT will be set in the outer header independent of the SA. Congestion along the route may then result in ECN-capable routers setting CE in the outer header. All packets arriving at the tunnel egress on this SA will appear to be case (b) errors, but should be processed according to whether CE was set. Therefore it is RECOMMENDED that packets violating the condition for case (b) above be dropped if CE is set to 1 in the outer header and forwarded if CE is 0 in the outer header.

An IPsec tunnel cannot provide protection against erasure of congestion indications or false reports of congestion based on flipping the value of the CE bit in packets for which ECT is set in the outer header. As described in Section 5, false reports of congestion are equivalent to dropping the packet, an action against which IPsec also provides no protection. On the other hand, erasure of congestion indications may impact the network and other flows in ways that would not be possible in the absence of ECN. It is important to note that erasure of congestion indications can only be performed to congestion indications placed by nodes within the tunnel; the copy of the CE bit in the inner header preserves congestion notifications from nodes upstream of the tunnel ingress. If erasure of congestion notifications is judged to be a security risk that exceeds the congestion management benefits of ECN, the security administrator should configure the appropriate tunnel SAs to forbid ECN usage in the outer header.

7. Issues Raised by Monitoring and Policing Devices

One possibility is that monitoring and policing devices (or more informally, ‘‘penalty boxes’’) will be installed in the network to monitor whether best-effort flows are appropriately responding to congestion, and to preferentially drop packets from flows determined not to be using adequate end-to-end congestion control procedures. [FF98] proposes three potential classifications for high-bandwidth flows in times in congestion: (1) flows that are not TCP-friendly, in that the arrival rate from that flow exceeds the arrival rate of a conformant TCP connection under the same conditions; (2) flows that are unresponsive, in that they do not decrease their arrival rate appropriately in response to an increase in congestion; and (3) flows using disproportionate bandwidth, defined as flows using a significantly larger share of bandwidth than other flows in times of high congestion. The methods of identifying and classifying flows to be in one of these three categories is outside the scope of this discussion.

[FF98] proposes that flows that are simply determined to be using disproportionate bandwidth could have their bandwidth restricted, in much the same way that a round-robin per-flow scheduling algorithm would restrict the bandwidth received by individual flows, while flows determined to be unresponsive or not TCP-friendly in times of congestion could have their bandwidth even more strongly reduced, as a concrete incentive to end nodes to use end-to-end congestion control.

For an ECN-capable flow, an ‘ideal’ penalty box at a router would be a device that, when it detected that a flow was not responding to ECN indications, would switch to dropping, instead of marking, those packets of a flow that would otherwise have been chosen to carry indications of congestion. In this way, these congestion indications could not be ‘erased’ later in the network, and at the same time there would be no change in the router’s treatment of packets of other flows. If a router determines that a flow is still not responding to congestion indications, when the congestion indications consist of packet drops, then the router could take whatever action it deems appropriate for that flow.

We recommend that any ‘‘penalty box’’ that detects a flow or an aggregate of flows that is not responding to end-to-end congestion control first change from marking to dropping packets from that flow, before taking any additional action to restrict the bandwidth available to that flow. Thus, initially, the router may drop packets in which the router would otherwise would have set the CE bit. This could include dropping those arriving packets for that flow that are ECN-Capable and that already have the CE bit set. In this way, any congestion indications seen by that router for that flow will be guaranteed to also be seen by the end nodes, even in the presence of malicious or broken routers elsewhere in the path. If we assume that the first action taken at any ‘‘penalty box’’ for an ECN-capable flow will be to drop packets instead of marking them, then there is no way that an adversary that subverts ECN-based end-to-end congestion control can cause a flow to be characterized as being non-cooperative and placed into a more severe action within the ‘‘penalty box’’.

The monitoring and policing devices that are actually deployed could fall short of the ‘ideal’ monitoring device described above, in that the monitoring is applied not to a single flow or to a single IPsec tunnel, but to an aggregate of flows. In this case, the switch from marking to dropping would apply to all of the flows in that aggregate, denying the benefits of ECN to the other flows in the aggregate also. At the highest level of aggregation, another form of the disabling of ECN happens even in the absence of monitoring and policing devices, when ECN-Capable RED queues switch from marking to dropping packets as an indication of congestion when the average queue size has exceeded some threshold.

7.1. Complications Introduced by Split Paths

If a router or other network element has access to all of the packets of a flow, then that router could do no more damage to a flow by altering the ECN field that it could by simply dropping all of the packets from that flow. However, in some cases, a malicious or broken router might have access to only a subset of the packets from a flow. The question is as follows: can this router, by altering the ECN field in this subset of the packets, do more damage to that flow than if it has simply dropped that set of the packets?

We will classify the packets in the flow as A packets and B packets, and assume that the adversary only has access to A packets. Assume that the adversary is subverting end-to-end congestion control along the path traveled by A packets only, by either falsely indicating ECN-Capability upstream of the point where congestion occurs, or erasing the congestion indication downstream. Consider also that there exists a monitoring device that sees both the A and B packets, and will "punish" both the A and B packets if the total flow is determined not to be properly responding to indications of congestion. Another key characteristic that we believe is likely to be true is that the monitoring device, before ''punishing'' the A&B flow, will first drop packets instead of setting the CE bit, and will drop arriving packets of that flow that already have the ECT and CE bits set. If the end nodes are in fact using end-to-end congestion control, they will see all of the indications of congestion seen by the monitoring device, and will begin to respond to these indications of congestion. Thus, the monitoring device is successful in providing the indications to the flow at an early stage.

It is true that the adversary that has access only to the A packets might, by subverting ECN-based congestion control, be able to deny the benefits of ECN to the other packets in the A&B aggregate. While this is unfortunate, this is not a reason disable ECN within an IPsec tunnel.

A variant of falsely reporting congestion occurs when there are two adversaries along a path, where the first adversary falsely reports congestion, and the second adversary 'erases' those reports. (Unlike packet drops, ECN congestion reports can be 'reversed' later in the network by a malicious or broken router.) While this would be transparent to the end node, it is possible that a monitoring device between the first and second adversaries would see the false indications of congestion. Given our recommendation in this document, before 'punishing' a flow for not responding appropriately to congestion, the router will first switch to dropping rather than marking as an indication of congestion, for that flow. When this includes dropping arriving packets from that flow that have the CE bit set, this ensures that these indications of congestion are being seen by the end nodes. Thus, there is no additional harm that we are able to postulate as a result of multiple conflicting adversaries.

8. Conclusions.

When ECN (Explicit Congestion Notification [RFC2481]) is used, it is desirable that congestion indications generated within an IPsec tunnel not be lost at the tunnel egress. We propose a minor modification to the IPsec protocol's handling of the ECN field during encapsulation and de-capsulation to allow flows that will undergo IPsec tunneling to use ECN.

Two options were proposed:
1) A preferred alternative, which is the full-functionality option as described in RFC 2481. This copies the ECT bit of the inner header to the encapsulating header. At decapsulation, if the ECT bit is set in the inner header, the CE bit on the outer header is ORed with the CE bit of the inner header to update the CE bit of the packet. 2) A limited-functionality option that does not use ECN inside the IPsec tunnel, by turning the ECT bit in the outer header off, and not

altering the inner header at the time of decapsulation.

We also proposed a new IPsec SA attribute to support negotiation of ECN support for tunnels between tunnel end-points and a new field in the Security Association database to indicate whether ECN is supported in tunnel mode on a SA.

We examined the consequence of modifications of the ECN field within the tunnel, analyzing all the opportunities for an adversary to change the ECN field. In many cases, the change to the ECN field is no worse than dropping a packet. However, we noted that some changes have the more serious consequence of subverting end-to-end congestion control. However, we point out that even then the potential damage is limited, and is similar to the threat posed by an end-system intentionally failing to cooperate with end-to-end congestion control.

We therefore believe that with these changes it is reasonable to use ECN with IPsec tunnels, as described in RFC 2481.

9. Acknowledgements

We thank Steve Bellovin and Vern Paxson for discussions of these matters.

10. References

[FF98] Floyd, S., and Fall, K., Promoting the Use of End-to-End Congestion Control in the Internet. Under submission, February 1998. URL "http://www-nrg.ee.lbl.gov/floyd/end2end-paper.html".

[RFC 2401] S. Kent, R. Atkinson, Security Architecture for the Internet Protocol, RFC 2401, November 1998.

[RFC2407] D. Piper, The Internet IP Security Domain of Interpretation for ISAKMP, RFC 2407, November 1998.

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

[RFC2481] K. Ramakrishnan, S. Floyd, A Proposal to add Explicit Congestion Notification (ECN) to IP, RFC 2481, January 1999.

11. Security Considerations

Security considerations have been addressed in the main body of the document.

AUTHORS' ADDRESSES

Sally Floyd
AT&T Center for Internet Research at ICSI (ACIRI)
Phone: +1 (510) 642-4274 x189
Email: floyd@aciri.org
URL: http://www-nrg.ee.lbl.gov/floyd/

David L. Black
EMC Corporation
42 South St.
Hopkinton, MA 01748
Phone: +1-508-435-1000 x75140
EMail: black_david@emc.com

K. K. Ramakrishnan
AT&T Labs. Research
Phone: +1 (973) 360-8766
Email: kkrama@research.att.com
URL: http://www.research.att.com/info/kkrama

This draft was created in February 1999.
It expires August 1999.