

Explicit Congestion Notification (ECN) for TCP-IP

Joseph Davies

When routers become congested to the point in which their incoming packet buffers fill and they begin dropping packets, the effect on the network is reduced bandwidth, an impact on loss or time-sensitive traffic flows, and possibly link idle time after the congestion occurs. Explicit Congestion Notification (ECN) for TCP/IP provides a way for routers to inform Transmission Control Protocol (TCP) peers that their buffers are filling due to congestion in the network. In response, TCP peers slow their transmission of data to help prevent packet losses. This article describes the need for ECN, how it works, and how ECN is supported in Windows Vista and Windows Server 2008.

Note: This article assumes in-depth knowledge of IP, TCP, and IP routing.

Introduction

Modern implementations of TCP treat the intermediate network between TCP peers as an opaque box. Packets containing TCP segments go into and out of the box. Sometimes the packets that go into the box are dropped. Because bit-level errors on today's digital and optical media are relatively rare, the designers of TCP made the assumption that a dropped packet is most likely due to congestion at a router a congested router's buffers for incoming packets have filled to capacity and the router is silently discarding incoming packets.

Although TCP detects that the packets containing TCP segments for a connection were dropped and retransmits them, recovering from dropped packets is an expensive process in terms of sending host TCP processing, retransmission of the dropped packet, and reduced throughput.

When a sending TCP peer detects a dropped packet, either through fast retransmit or because the retransmission timeout on the segment expires, it retransmits the segment. The sending TCP peer then reduces the send window (the number of segments that it can send before waiting for an acknowledgement) and performs the slow start and congestion avoidance algorithms (RFC 2581). This immediately lowers the transmission rate of the sender to allow time for the routers to clear their congested buffers. The sender gradually increases its send window back to the size of the send window before the congestion occurred.

Although packet drops due to congested routers are an unfortunate occurrence, they do not negatively impact bulk data transfers other than the additional time required to retransmit the dropped segments and gradually increase the transmission rate. The slow start and congestion avoidance algorithms work well for time-insensitive, bulk data traffic. However, the TCP method for dealing with dropped packets does not work as well for interactive, loss-sensitive, or time-sensitive traffic.

Another issue with router congestion is the effect that congestion has on multiple data flows. When a router begins dropping incoming packets, it typically does not distinguish one data flow from another. When multiple TCP flows have packet drops, the senders of all of those flows reduce their transmission rate. Depending on how quickly the router clears its congested buffers, the multiple TCP data flows that had packet drops might still be gradually increasing their sending rate. This can result in the router and its links being underutilized until all of the TCP data flows are sending at their pre-congestion transmission rates. The router goes from a congested state to an underutilized state.

The issues of lowered throughput through retransmission and lower link utilization after congestion are the consequences of attempting to manage congestion only from the sending host, in which the only congestion indicator is dropped packets. To prevent the problems associated with dropped packets due to congested routers, the designers of TCP/IP have created a new set of standards for both hosts and routers. These standards describe active queue management (AQM) on IP routers (RFC 2309) to allow the router to monitor that state of its forwarding queues and provide a mechanism to allow routers to report to sending hosts that congestion is occurring, allowing the sending hosts to lower their transmission rate before the router begins dropping packets. The router

Explicit Congestion Notification (ECN) for TCP-IP

Joseph Davies

reporting and host response mechanism is known as Explicit Congestion Notification (ECN) (RFC 3168).

When congestion occurs, sending hosts must still lower their transmission rates. However, by avoiding packet losses, sending hosts no longer incur the packet processing required to retransmit dropped packets and time and loss-sensitive packet flows are not impacted as severely during congestion.

Explicit Congestion Notification

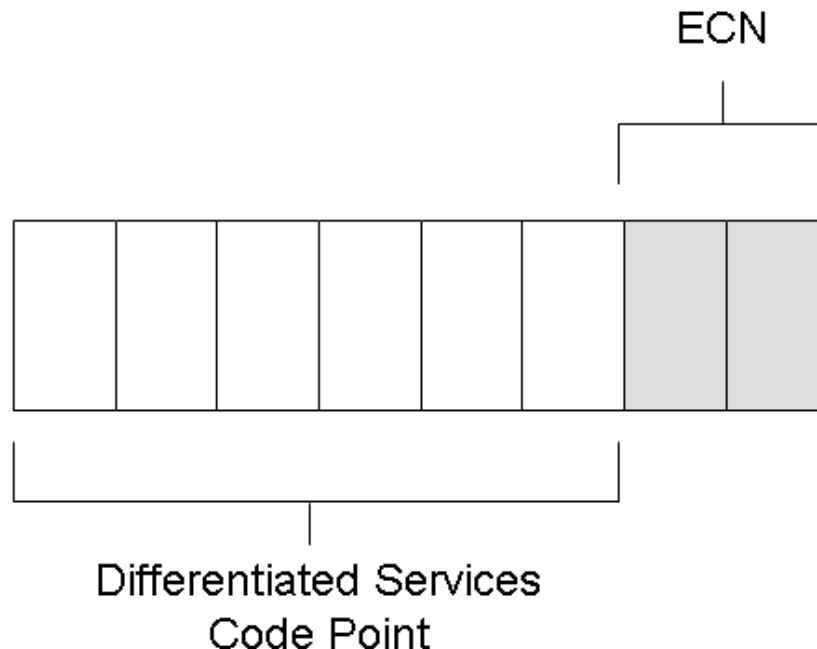
ECN support for TCP/IP uses unused bits in both the IP and TCP headers.

- At the Internet layer (for IP), a sending host must be able to indicate that it is capable of performing ECN and a router must be able to indicate that it is experiencing congestion when forwarding a packet.
- At the Transport layer (for TCP), TCP peers must indicate to each other that they are ECN-capable. A receiving peer must be able to inform the sending peer that it has received a packet from a router experiencing congestion. The sending peer must be able to inform the receiving peer that it has received the congestion indicator from the receiving peer and has reduced its transmission rate.

The following sections describe the details of ECN support in IP and TCP.

ECN Support in IP

The 8-bit Type of Service (TOS) field in the IP header was originally defined in RFC 791 to indicate delivery precedence, delay, throughput, reliability, and cost characteristics of a packet for non-default delivery by routers. The TOS field was redefined in RFC 2474 as the Differentiated Services (DS) field containing a 6-bit Differentiated Services Code Point (DSCP) value and two unused bits. The DSCP value indicates a delivery priority that corresponds to queues previously configured in the routers of an intranet. ECN support in IP uses the two unused bits of the RFC 2474-defined TOS field. The following figure shows the new definition of the DS field with ECN.



Explicit Congestion Notification (ECN) for TCP-IP

Joseph Davies

The two unused bits in the RFC 2474-defined DS field are defined in RFC 3168 as the ECN field, which has the following values:

- 00 The sending host does not support ECN.
- 01 or 10 The sending host supports ECN.
- 11 Congestion has been experienced by a router.

An ECN-capable host sends its packets with the ECN field set to 01 or 10. For packets sent by ECN-capable hosts, if a router in the path is ECN-capable and is experiencing congestion, it sets the ECN field to 11. If the ECN field has been set to 11, downstream routers in the path to the destination do not modify its value.

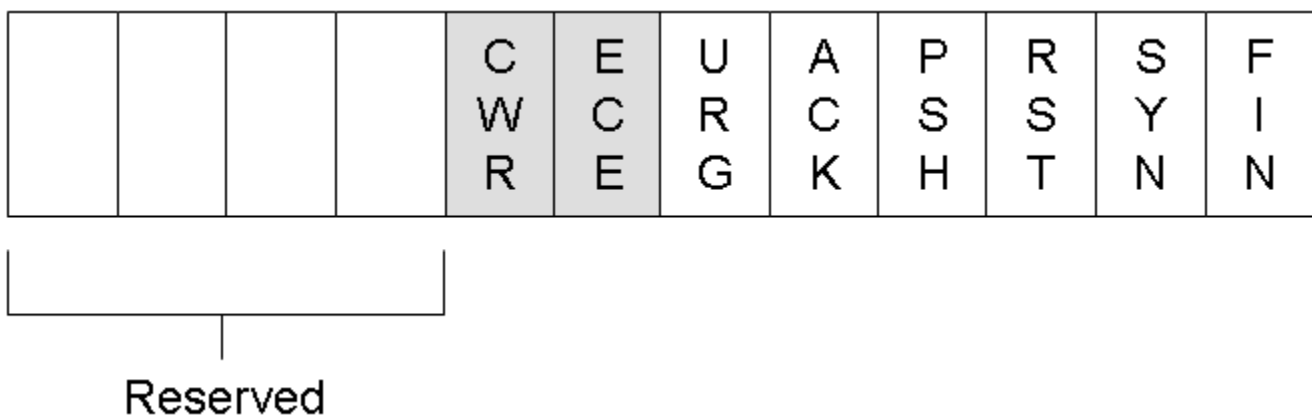
ECN Support in TCP

When an IP packet ECN field is set to 11 by a router, the receiver is informed of the congestion in the path, but not the sender. ECN uses the TCP header to indicate to the sender that the network is experiencing congestion and to indicate to the receiver that the sender has received the congestion indication from the receiver and has lowered its transmission rate.

ECN support in TCP uses two bits in the TCP header that were previously defined as reserved. The two new flags defined for ECN support are the following:

- **ECE:** The ECN-Echo (ECE) flag is used to indicate that a TCP peer is ECN-capable during the TCP 3-way handshake and to indicate that a TCP segment was received on the connection with the ECN field in the IP header set to 11. For information about the TCP 3-way handshake, see RFC 793.
- **CWR:** The Congestion Window Reduced (CWR) flag is set by the sending host to indicate that it received a TCP segment with the ECE flag set. The congestion window is an internal variable maintained by TCP to manage the size of the send window.

The following figure shows the location of the ECE and CWR flags relative to the other flags in the TCP header. For information about the additional flags in the TCP header, see RFC 793.



When two ECN-capable TCP peers establish a TCP connection, they exchange Synchronize (SYN), SYN-Acknowledgement (SYN-ACK), and ACK segments. For ECN-capable TCP peers, the SYN segment has both the ECE and CWR flags set. The SYN-ACK segment has the ECE flag set and the CWR flag cleared.

Explicit Congestion Notification (ECN) for TCP-IP

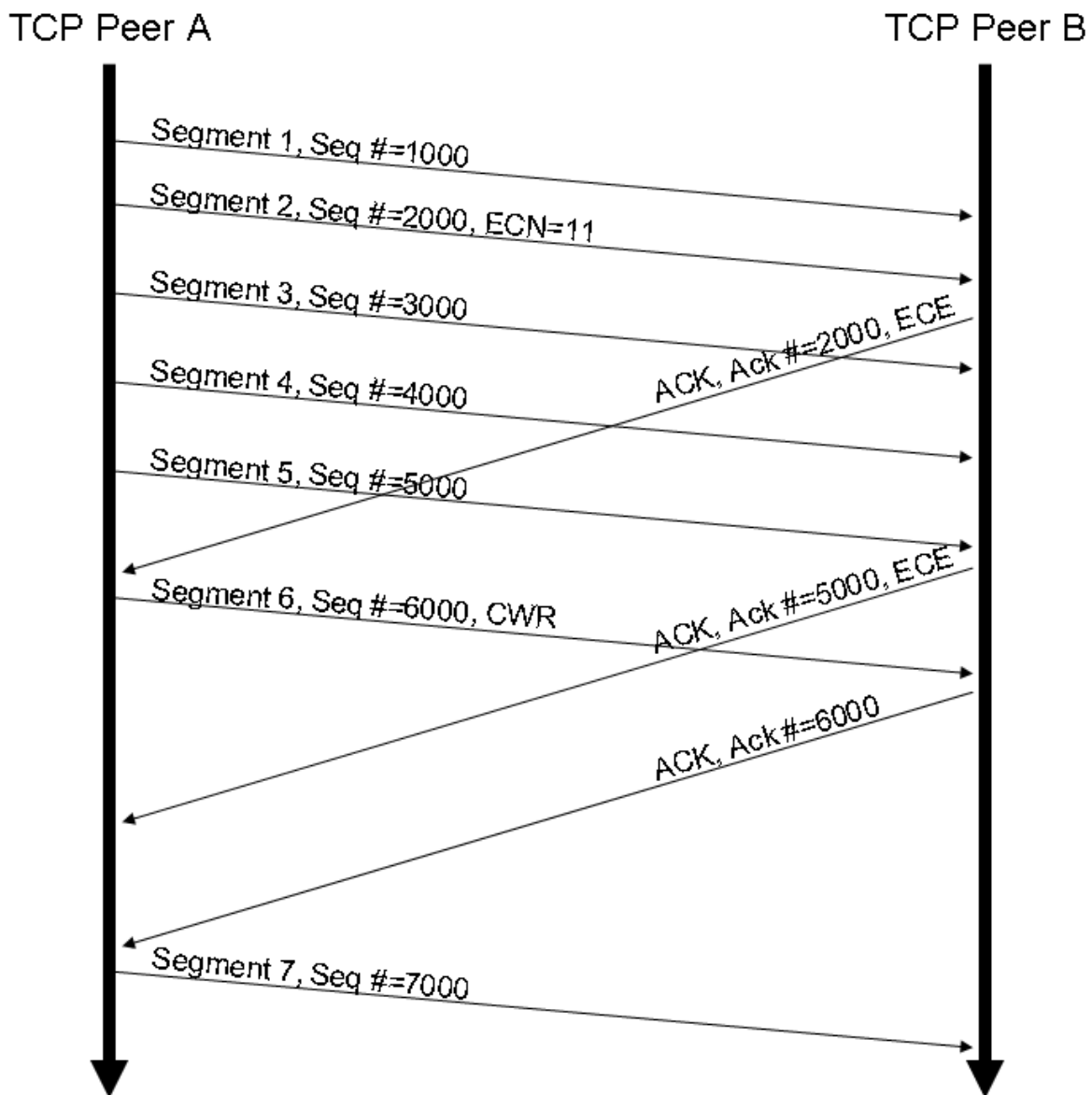
Joseph Davies

An ECN-capable host sends TCP segments for an ECN-enabled TCP connection with the ECN field in the IP header set to either 10 or 01. An ECN-capable router that is experiencing congestion sets the ECN field in the IP header to 11. When a receiving TCP peer sends an ACK that includes the data of a received TCP segment that had the ECN field set to 11, it sets the ECE flag in the TCP header and continues setting the ECE flag in subsequent ACKs.

When the sending host receives the ACK with the ECE flag set, it behaves as though a packet was dropped by reducing its send window and running the slow start and congestion avoidance algorithms. For the next segment, the sender sets the CWR flag. Upon receipt of the new segment with the CWR flag set, the receiver stops setting the ECE flag in subsequent ACKs.

ECN Example

The following figure shows an example of a TCP connection between ECN-capable TCP peers that experiences congestion by an ECN-capable router.



Explicit Congestion Notification (ECN) for TCP-IP

Joseph Davies

In this example, TCP Peer A is sending data to TCP Peer B. TCP Peer A sends Segments 1 through 5. Segment 2 is forwarded by an ECN-capable router that is experiencing congestion, which sets the ECN field in the IP header to 11. When TCP Peer B receives this segment, it sends ACKs with the ECE flag set. When TCP Peer A receives the first ACK with the ECE flag set, it lowers its transmission rate and sends its next segment (Segment 6) with the CWR flag set. Upon receipt of the Segment 6 with the CWR flag set, TCP Peer B sends subsequent ACKs with the ECE flag cleared.

For information about the behavior of ECN for different variations of TCP data flow, see RFC 3168.

ECN Support in Windows

Windows Vista supports ECN but it is disabled by default. To enable ECN support, use the netsh interface tcp set global ecncapability=enabled command. Because ECN is using bits in the IP and TCP headers that were previously defined as unused or reserved, intermediate network devices such as routers and firewalls might silently discard packets when the ECN fields are set to non-zero values. To ensure that ECN-marked TCP/IP traffic will not be dropped from your network, survey your networking equipment and perform the appropriate configuration or upgrades to ensure that ECN-marked packets are not discarded.