Internet Engineering Task Force
INTERNET DRAFT
draft-ietf-tsvwg-tcp-ecn-00.txt

Sally Floyd
ACIRI
K. K. Ramakrishnan
TeraOptic Networks

November 2000
Expires: May 2001

TCP with ECN: The Treatment of Retransmitted Data Packets

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

This document makes recommendations for the use of ECN with retransmitted data packets, for an ECN-capable TCP connection. This document supplements RFC 2481 [RFC2481], which did not address the issue of retransmitted data packets. This document recommends that for ECN-capable TCP implementations, the ECT bit (ECN-Capable Transport) in the IP header SHOULD NOT be set on retransmitted data packets, and that the TCP data receiver SHOULD ignore the ECN field on arriving data packets that are outside of the receiver's current window. This is for greater security against denial-of-service attacks.

In addition, this document recommends that the CWR bit (Congestion Window Reduced) in the TCP header SHOULD NOT be set on retransmitted packets. When the TCP data sender is ready to set the CWR bit after reducing the congestion window, it SHOULD set the CWR bit on the first new data packet that it transmits.

1. Introduction

RFC 2481, which describes both the TCP and IP semantics for ECN, does not make any special mention of retransmitted TCP packets. It is important to make special mention of the use of ECN with retransmitted packets, especially the setting of the ECT bit on retransmitted packets. If TCP is allowed to set the ECT bit on retransmitted data packets, this could open the door for denial-of-service attacks.

In particular, an attacker capable of spoofing the IP source address could send data packets with arbitrary sequence numbers, with both the ECT and CE bits set in the IP header. On receiving this spoofed data packet, the TCP data receiver would determine that the data does not lie in the current receive window, and return a duplicate acknowledgement. We define an out-of-window packet at the TCP data receiver as a data packet that lies outside the receiver's current window. On receiving an out-of-window packet, the TCP data receiver has to decide whether or not to treat the CE bit in the packet header as a valid indication of congestion, and therefore whether to return ECN-Echo indications to the TCP data sender. If the TCP data receiver ignored the CE bit in an out-of-window packet, then the TCP data sender would not receive this possibly-legitimate indication of congestion, resulting in a violation of end-to-end congestion control. On the other hand, if the TCP data receiver honors the CE indication in the out-of-window packet, and reports the indication of congestion to the TCP data sender, then the malicious node that created the spoofed, out-of-window packet has successfully ``attacked'' the TCP connection by forcing the data sender to unnecessarily reduce (halve) its congestion window. To prevent such a denial-of-service attack, Section 2 of this document recommends that a legitimate TCP data sender SHOULD NOT set the ECT bit on retransmitted data packets, and that the TCP data receiver SHOULD ignore the CE bit on out-of-window packets.

2. ECN and Retransmitted Data Packets

In this document we assume an environment where it is possible for a malicious host to spoof the legitimate IP source address of a TCP connection, and to send a data packet with the spoofed source address along with the ECT and CE bits set. In order to protect itself against a denial-of-service attack, the TCP connection has to refrain

from halving its congestion window in response to such a spoofed data packet.

There are several possible ways that a TCP connection could protect itself from such a denial-of-service attack:

(1) Not using ECN on retransmits: The TCP receiver could ignore the CE bit on out-of-window packets, knowing that a conformant sender would not set the ECT bit on retransmitted packets.

(2) Ignoring ECN on out-of-window packets: The TCP receiver could ignore the CE bit on out-of-window packets, knowing that a conformant sender would not set the ECT bit on retransmitted packets.

(3) Ignoring ECN on significantly out-of-window packets: The TCP receiver could ignore the CE bit on packets far outside the current window, while a conformant sender could be allowed to set the ECT bit on retransmitted packets.

(4) Verifying out-of-window ECN packets: The TCP receiver could report to the sender the sequence numbers of an out-of-window data packet with the CE bit set, and the sender could halve its congestion window only if it had in fact recently sent this packet.

In this document we discuss each of these four proposals, and explain why we follow the first proposal, to not use ECN on retransmitted data packets (by not setting the ECT bit on these packets), for ECN-Capable TCP implementations.

2.1. **Option 1:** Not Using ECN on Retransmits.

This document recommends Option 1, not using ECN on retransmitted packets. This approach is simple, straightforward, and protects against ECN-based denial-of-service attacks. (This assumes that the attacker is unable to guess the initial sequence number (ISN) of a TCP connection, and therefore is unlikely to guess a sequence number for a spoofed packet that is within the current window.)

The drawback of Option 1 is that it denies ECN protection for retransmitted packets. However, for an ECN-capable TCP connection in a fully-ECN-capable environment with mild congestion, packets should rarely be dropped due to congestion in the first place, and so instances of retransmitted packets should rarely arise. If packets are being retransmitted, then there are already packet losses (from corruption or from congestion) that ECN has been unable to prevent.

We note that, with Option 1, if the router sets the CE bit for an ECN-capable data packet within a TCP connection, then the TCP connection is guaranteed to receive that indication of congestion, or to receive some other indication of congestion within the same window of data, even if this packet is dropped or reordered in the network. We consider two cases, when the packet is later retransmitted, and when the packet is not later retransmitted.

In the first case, if the packet is either dropped or delayed, and at some point retransmitted by the data sender, then the retransmission is a result of a Fast Retransmit or a Retransmit Timeout for either that packet or for some prior packet in the same window of data. In this case, because the data sender already has retransmitted this packet, we know that the data sender has already responded to an indication of congestion for some packet within the same window of data as the original packet. Thus, even if this retransmitted packet is dropped in the network, or is delayed, with the CE bit set, and is later ignored by the data receiver as an out-of-window packet, this is not a problem, because the sender has already responded to an indication of congestion for that window of data. Communicating the indication of congestion with the CE bit for this retransmitted packet (if that indication is provided to the sender within the same window of data as a previously dropped packet) would not have resulted in any further reduction of the window by the sender.

In the second case, if the packet is never retransmitted by the data sender, then this data packet is the only copy of this data received by the data receiver, and therefore arrives at the data receiver as an in-window packet, regardless of how much the packet might be delayed or reordered. In this case, if the CE bit is set on the packet within the network, this will be treated by the data receiver as a valid indication of congestion.

2.2. **Option 2:** Ignoring ECN on Out-of-window Packets?

A second option for protecting against ECN-based denial-of-service attacks would be for the TCP data sender to be allowed to set the CE bit on retransmitted packets, but for the TCP data receiver to ignore the CE bit on out-of-window data packets. However, this would have the unfortunate consequence of weakening the effectiveness of ECN-based end-to-end congestion control (by carrying over to ECN some of the existing weaknesses of packet drops as indications of congestion).

We will say that a retransmitted TCP packet is ``necessarily-retransmitted'' if the retransmission is the only copy of this data received at the TCP receiver, and ``unnecessarily-retransmitted'' otherwise. For TCP, drops of unnecessarily-retransmitted data

packets are not detected as indications of congestion by the TCP end nodes. That is, if a necessarily-retransmitted TCP packet is dropped, then the packet loss is detected by the TCP receiver, and TCP reduces its sending rate. In contrast, if an unnecessarily-retransmitted TCP packet is dropped, then that loss is not detected by the TCP receiver, and TCP does not reduce its sending rate.

Option 2, of ignoring ECN on out-of-window packets, would effectively prevent ECN-based denial-of-service attacks using retransmitted packets. At the same time, allowing the data receiver to ignore ECN information on out-of-window packets would carry over to ECN the weaknesses of packet drops as indications of congestion: that packet drops of unnecessarily-retransmitted packets are not detected as indications of congestion by the TCP end hosts. With Option 2, a necessarily-retransmitted TCP packet would be in-window when received at the TCP receiver, and ECN indications in the packet header would be treated normally by the TCP receiver. However, an unnecessarily-retransmitted TCP packet would most likely be out-of-window when received at the TCP receiver, and therefore, with Option 2, any ECN information in the packet header would be ignored. Setting the ECT bit on unnecessarily-retransmitted TCP packets has the potential that routers would mark rather than drop such packets. A receiver ignoring the CE bit on such a packet would result in not communicating the congestion indication to the TCP sender, making the ECN information as unreliable as packet drops for unnecessarily-retransmitted packets. However, it is not sufficient for ECN to be merely as reliable as packet losses as indications of congestion. An attempt by a congested router to avoid dropping the unnecessarily-retransmitted packet and provide an indication of congestion via ECN would be negated by Option 2. With Option 2, a transport would communicate that it is ECN-capable by setting the ECT bit, but would ignore the CE bit. In addition to violating the semantics of ECN, this could also have an impact on other flows, both in terms of fairness and the possible congestion experienced by such other flows.

The principle has already been established for TCP that the ECT bit should not be set on a packet if there will be no viable congestion-control response by the transport to a router setting the CE bit in the packet header. Consider the example of pure acknowledgement (ACK) packets. TCP does not have effective, loss-based end-to-end congestion control for ACK packets, that is, packets that don't contain any data. If a pure ACK packet in a TCP connection is dropped, the TCP connection does not respond by reducing its sending rate along that path. Because current TCP receivers have no mechanisms for reducing traffic on the ACK-path in response to congestion notification, RFC 2481 specifies that the ECT bit should not be set on pure ACK packets.

Therefore, our view is that Option 2 is not a viable option for preventing ECN-based denial-of-service attacks on a connection.

2.3. **Option 3: Ignoring ECN on significantly out-of-window packets:**

With Option 3, the TCP data sender would be allowed to set the CE bit on retransmitted packets, and the TCP connection would protect itself from ECN on spoofed packets by checking the sequence numbers of out-of-window packets that arrive with the CE bit set. Consider a data receiver that has acknowledged data up to and including sequence number N, and has a window of W bytes. If the out-of-window data is contained within the sequence number range (N-W, N] (modulo the proper amount), then the CE bit on the out-of-window packet is treated as a valid indication of congestion, and otherwise the CE bit on the out-of-window packet is ignored. This is based on the view that the receiver considers the range (N-W, N] to represent a range of data that could plausibly result from retransmissions from the legitimate data source.

Option 3 would make it somewhat easier for a malicious host to guess a sequence-number range for which a CE bit would be treated as a valid indication of congestion, and therefore would make an ECN-based denial-of-service attack somewhat more likely to be successful. However, Option 3 would still offer a significant protection against denial-of-service attacks because the malicious host has to guess the appropriate sequence number range.

However, Option 3 introduces some additional complexity at the TCP data receiver, and would not necessarily result in the CE bit being respected on all data packets actually sent by the legitimate data sender. While Option 3 has the benefit over Option 1 of allowing the sender to set the ECT bit on retransmitted packets, the benefit does not seem worth the additional cost at the data receiver.

2.4. **Option 4:** Verifying out-of-window ECN packets?

With Option 4, the TCP data sender would be allowed to set the CE bit on retransmitted packets, and the TCP connection would protect itself from ECN on spoofed packets by having the data receiver report to the data sender the sequence numbers of the data in out-of-window packets with the CE bit set. Using this sequence-number information, the data sender could verify that it had actually sent this retransmitted packet before honoring the ECN indication of congestion. If the data sender determined that this packet was in fact from another host that had spoofed its IP address, then the data sender could ignore this indication of congestion.

At first glance this seems simple, but Option 4 would actually turn

into something rather complicated. The ECN-Echo information is carried from the data receiver to the data sender not just in one ACK packet, but possibly in a number of successive ACK packets. In addition, the data receiver could receive multiple data packets with the CE bit set, which are treated by the data sender as a single instance of congestion. The benefits of Option 4 do not seem likely to be worth the extra complexity that would be entailed. In addition, this would require significant additional changes to the header and to the standardization process for such use.

In summary, this document recommends Option 1, that the ECT bit SHOULD NOT be set on retransmitted data packets, and that the TCP data receiver SHOULD ignore the ECN field on out-of-window data packets.

3. Setting the CWR bit.

RFC 2481 says the following regarding the setting of the CWR bit: ``When an ECN-Capable TCP reduces its congestion window for any reason (because of a retransmit timeout, a Fast Retransmit, or in response to an ECN Notification), the TCP sets the CWR flag in the TCP header of the first data packet sent after the window reduction. If that data packet is dropped in the network, then the sending TCP will have to reduce the congestion window again and retransmit the dropped packet. Thus, the Congestion Window Reduced message is reliably delivered to the data receiver."

However, as noted earlier in this document, if a retransmitted data packet is dropped in the network, the sending TCP does not necessarily respond by reducing its congestion window. Therefore, the CWR bit SHOULD NOT be set on retransmitted packets. Instead, when the TCP data sender is ready to set the CWR bit after reducing the congestion window, it SHOULD set the CWR bit on the first *new* data packet that it subsequently transmits.

4. ECN and window probes.

When the TCP data receiver advertises a zero window, the TCP data sender sends window probes to determine if the receiver's window has increased. Window probe packets do not contain any user data except for the sequence number, which is a byte. Because window probes use the exact sequence numbers, they cannot be easily spoofed in denial-of-service attacks. Therefore, if a window probe arrives with ECT and CE set, then the receiver SHOULD respond to the ECN indications.

At the same time, if a window probe packet is dropped in the network, this loss is not detected by the receiver. Therefore, the TCP data sender SHOULD NOT set either the ECT or CWR bits on window probe

packets.

5. Conclusions

This document recommends that for ECN-capable TCP implementations, the ECT bit SHOULD NOT be set on retransmitted data packets, and that the TCP data receiver SHOULD ignore the ECN field on out-of-window data packets. This is for greater security against denial-of-service attacks.

In addition, this document recommends that the CWR bit (Congestion Window Reduced) in the TCP header SHOULD NOT be set on retransmitted packets. When the TCP data sender is ready to set the CWR bit after reducing the congestion window, it SHOULD set the CWR bit on the first *new* data packet that it transmits.

Finally, this document recommends that a TCP data sender SHOULD NOT set either the ECT or CWR bits on window probe packets.

6. Acknowledgements

This note resulted from email discussions with a number of people, including Alexey Kuznetsov, Jamal Hadi-Salim, and Venkat Venkatsubra.

7. References

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

8. Security Considerations

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

AUTHORS' ADDRESSES

K. K. Ramakrishnan
TeraOptic Networks
Phone: +1 (408) 666-8650
Email: kk@teraoptic.com

Sally Floyd
Phone: +1 (510) 666-2989
ACIRI
Email: floyd@aciri.org
URL: http://www.aciri.org/floyd/

This draft was created in November 2000.
It expires May 2001.