Is Explicit Congestion Notification usable with UDP?

Stephen McQuistin
School of Computing Science
University of Glasgow
sm@smcquistin.uk

Colin Perkins
School of Computing Science
University of Glasgow
csp@csperkins.org

ABSTRACT

We present initial measurements to determine if ECN is usable with UDP traffic in the public Internet. This is interesting because ECN is part of current IETF proposals for congestion control of UDP-based interactive multimedia, and due to the increasing use of UDP as a substrate on which new transport protocols can be deployed. Using measurements from the author's homes, their workplace, and cloud servers in each of the nine EC2 regions worldwide, we test reachability of 2500 servers from the public NTP server pool, using ECT(0) and not-ECT marked UDP packets. We show that an average of 98.97% of the NTP servers that are reachable using not-ECT marked packets are also reachable using ECT(0) marked UDP packets, and that ~98% of network hops pass ECT(0) marked packets without clearing the ECT bits. We compare reachability of the same hosts using ECN with TCP, finding that 82.0% of those reachable with TCP can successfully negotiate and use ECN. Our findings suggest that ECN is broadly usable with UDP traffic, and that support for use of ECN with TCP has increased.

Categories and Subject Descriptors

C.2.5 [**Local and Wide-Area Networks**]: Internet

Keywords

Explicit congestion notification; network measurement; UDP

1. INTRODUCTION

Explicit congestion notification (ECN) is a mechanism that allows Internet routers to signal the presence of congestion to end systems without packet drops. The current ECN standard for TCP/IP [10] was published in 2001, and has seen wide implementation, but only moderate use, in part due to concerns about compatibility with firewalls and other middleboxes. Recent work [14] has eased those concerns, and use of ECN is growing. With this growth, we are now starting to see broader interest in using ECN, including use with transport protocols that layer above UDP.

ECN can be used with UDP to support interactive multimedia systems, such as those using the WebRTC framework [5]. WebRTC uses RTP [13] over UDP for media transport, and RTP extensions to support ECN feedback in this environment have recently been defined [15]. Congestion control algorithms for interactive video using RTP over UDP are under development in IETF, and one of the candidates, NADA [17], makes extensive use of ECN. This is desirable for interactive video, since ECN support in the network allows for lower queue occupancy, hence lower latency, and because the ability to react to congestion without packet loss avoids visible disruption to the video, improving the user experience.

Other cases where ECN is potentially helpful for UDP-based transports include congestion feedback for tunnelled pseudo-wire traffic, and when UDP is used as a substrate for new transport protocol development. Examples of the latter include Google's QUIC protocol [12], SPUD [3], and various proposals for IP stack evolution discussed at the recent IAB workshop on Stack Evolution in a Middlebox Internet [4].

In this paper, we present results of an initial measurement study to determine the impact of ECN on UDP reachability. Based on a study of the servers in the NTP pool, we show that the presence of ECN marks has a small, but non-zero, impact on reachability of UDP servers, with an average of 98.97% of the NTP servers that are reachable without ECN also being reachable when ECN is used, and ~98% of network hops passing ECN marks. For comparison, we find that 82.0% of the servers that are reachable using TCP will successfully negotiate ECN.

To the best of our knowledge, ours is the first study to measure the impact of ECN on reachability of UDP servers, and to offer a comparison with the ability to negotiate ECN for TCP connections. Our results, should they be replicated in larger studies, demonstrate that ECN is generally safe to enable for UDP traffic on the Internet. Furthermore, we show that ECN is increasingly usable with TCP, with greater willingness of TCP servers to negotiate and use ECN.

We structure the remainder of this paper as follows. We begin by reviewing the ECN standards and their use with different transport protocols in Section 2. Our experimental methodology is discussed in Section 3, and we present our measurement results in Section 4. We discuss related work in Section 5, and conclude in Section 6.

2. BACKGROUND

The Internet relies on packet loss as a congestion signal. Routers queue packets on their outgoing links, and congestion results in queue overflow and packet loss. The transport detects this loss, and sends feedback to the sender to reduce its transmission rate, completing the feedback loop. The addition of ECN allows routers to mark packets as a signal that queues are growing, indicating the presence of congestion before it becomes necessary to discard packets. The receiver detects marked packets, and informs the sender, which reacts to the indication as it would react to loss.

ECN takes two bits from the IP header to indicate if a packet belongs to an ECN capable transport (ECT) flow (00 = not ECT, 01 = ECT(1), and 10 = ECT(0), where ECT(0) and ECT(1) are equivalent). Routers that receive packets marked ECT(0) or ECT(1), and that are experiencing congestion, remark some of those packets by setting the ECT bits to 11 (ECN-CE), indicating congestion on the path. When ECN is used with TCP transport, that feedback is provided by use of two previously reserved bits in the TCP header: ECE (ECN-Echo) and CWR (Congestion Window Reduced). On receipt of an IP packet marked ECN-CE, TCP sets the ECN-Echo bit in the corresponding ACK packet. The sender, on receipt of an ACK with ECN-Echo set, reacts to congestion as if the packet were dropped, and sets the CWR flag in the TCP header of its next outgoing TCP segment to acknowledge its response to the congestion. Since ECN for TCP uses two previously reserved bits of the TCP header, and requires active participation from the receiver, it must be negotiated before use. The initiator of a TCP connection signals its desire to use ECN by setting both ECE and CWR on the SYN packet; if the receiver also understands and desires to use ECN, it will set ECE on the SYN-ACK.

UDP provides no feedback, so cannot directly be used with ECN. Rather, ECN is used in the context of a higher layer transport that runs over UDP and provides the necessary feedback. One such protocol is RTP [13], for which ECN feedback is defined in [15]. The use of ECN with RTP is negotiated using a non-RTP signalling channel, such as SIP [11] or WebRTC [5], and both endpoints need to agree to its use before data is sent with ECT markings. Other transports layered on UDP can support ECN in a similar way, with an initial ECN capability negotiation phase while the communication session is being set-up, before ECT-marked UDP packets are sent.

Since ECN has not been used with UDP until recently, it is possible that some firewalls, or other middleboxes, will regard the presence of ECN marked UDP as suspicious, and discard the traffic. The goal of this work is to determine the extent to which that happens, and understand whether ECN is usable with UDP in the Internet.

3. METHODOLOGY

To determine if ECN affects reachability when using UDP over the public Internet, we need a set of publicly available UDP-based servers to test against. To allow us to compare against TCP usability with ECN, it's desirable if those servers are also reachable using TCP. A set of servers that meet these constraints are the network time protocol (NTP) pool servers (DNS servers could also be used, and may be more representative of core infrastructure; we believe NTP pool servers better represent servers for other UDP applications).

NTP is a UDP-based client-server protocol that can be used for precision timekeeping. The NTP server pool is a worldwide, volunteer operated, virtual cluster of NTP servers that provide a publicly available time service. Servers in the pool are assumed to have stable IP addresses, and clients lookup an appropriate server with a DNS query for the `pool.ntp.org` domain. The pool operates round-robin DNS that returns a different answer every few minutes, to ensure clients are load-balanced across the servers in the pool. In addition to the UDP-based NTP service, each host in the pool is encouraged to run a web server providing a redirect to the main NTP pool website at `www.pool.ntp.org`. This combination gives us access to a worldwide pool of servers, accessible using both UDP and TCP, against which we can test ECN reachability.

To discover servers in the NTP pool, we wrote a script to perform a DNS query for `pool.ntp.org` and each of its country- and region-specific sub-domains in turn, with a one second gap between each query. This script was run at approximately ten minute intervals for a period of several weeks in March/April 2015, and discovered the addresses of a total of 2500 servers out of the NTP pool. These servers form the measurement targets in our study.

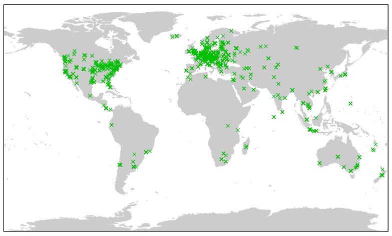


Figure 1: Geographic locations of NTP pool servers

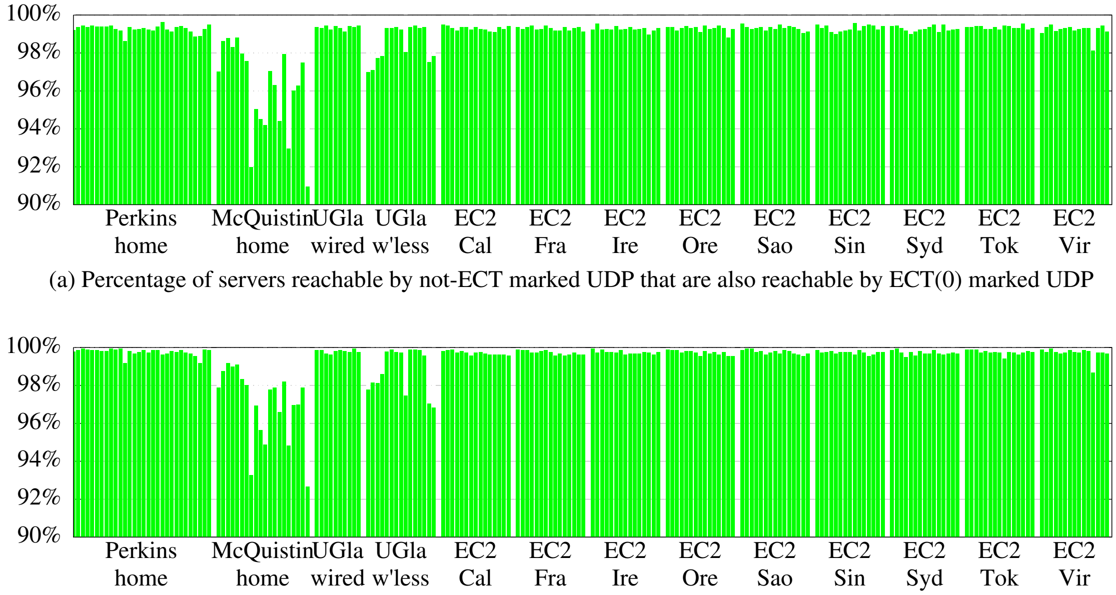
Region	NTP Server Count
Africa	22
Asia	190
Australia	68
Europe	1664
North America	522
South America	32
Unknown	2
Total	2500

Table 1: Geographic distribution of NTP pool servers

The approximate locations of these NTP servers were found using the MaxMind GeoLite2 City database, as of 25 April 2015, and are shown in Figure 1 and summarised in Table 1. The servers under study are distributed around the world, albeit with strongest coverage in Europe and North America, moderate coverage in parts of Asia and Australia, and only limited coverage in South America and Africa. While broader coverage in those regions would be desirable, we believe this set of servers does have sufficient reach to give meaningful results regarding ECN usability.

We conduct measurements against each discovered server, to evaluate its reachability with UDP (NTP) and TCP (HTTP), with and without ECN. In total, we perform 210 traces, where each trace tests both protocols, with and without the use of ECN, against each of the 2500 servers. Traces were collected from the authors' homes (connected via two different UK ISPs), from the University of Glasgow (using both wired and wireless connections), and using virtual machines running on each of the nine regions of the Amazon EC2 service (N. Virginia, Oregon, N. California, Ireland, Frankfurt, Singapore, Tokyo, Sydney, and Sao Paolo). These measurement points give broad geographical reach, albeit from a small number of networks. The data was collected in two batches: initial traces from the authors' homes and the University of Glasgow wireless in April/May 2015, with further traces from those locations and from EC2 in July/August 2015. Traces were collected using a custom measurement application. For each of the 2500 servers in turn, this application probes reachability for UDP and TCP based services, with and without use of ECN.

To probe reachability of UDP based services, our measurement application implements a custom NTP client. An NTP request is sent in a not-ECT marked UDP packet, and the response, if any, is recorded using a parallel `tcpdump` session. If no response is received, the request is retransmitted up to five times, with a one second timeout for each retransmission. If an NTP response is received after any request, we mark the server as reachable without ECN; otherwise it is marked as unreachable after five requests have



(a) Percentage of servers reachable by not-ECT marked UDP that are also reachable by ECT(0) marked UDP

(b) Percentage of servers reachable by ECT(0) marked UDP that are also reachable by not-ECT marked UDP

Figure 2: Reachability of NTP servers using UDP and UDP with ECT(0) marks for all traces, one bar per trace

timed out. The process is then repeated using NTP requests sent in an ECT(0) marked UDP packet, to determine reachability of that server with ECN (we use ECT(0) rather than ECT(1), to match the typical marking used with ECN for TCP). This allows us to check if the path from client to server passes ECT(0) marked UDP packets. Since we test against unmodified NTP servers, we cannot probe the return path from server to client.

To test reachability using TCP, we make an HTTP GET request for the root page of the server, without attempting to negotiate ECN, and record if the server responds to HTTP, and what HTTP response is received. We repeat the HTTP request, this time with ECN enabled, using an ECN-setup SYN packet to negotiate the use of ECN for the HTTP connection to the server. A parallel `tcpdump` session records the response, and is used to determine whether the returned SYN-ACK packet is an ECN-setup SYN-ACK packet.

Each of the four measurements (UDP, UDP with ECN, TCP, and TCP with ECN) is done for each of the 2500 servers in turn, to form a complete trace. Our data set comprises 210 such traces. It is available at http://dx.doi.org/10.5525/gla.researchdata.207.

4. RESULTS

We present four sets of results. Section 4.1 reports on the impact of ECN on reachability of UDP servers, and is supplemented in Section 4.2 with a discussion of whether and where ECN marks are stripped from UDP packets. Section 4.3 presents results on the fraction of TCP servers that will negotiate ECN, and we compare reachability between the two protocols in Section 4.4.

4.1 Reachability using ECN with UDP

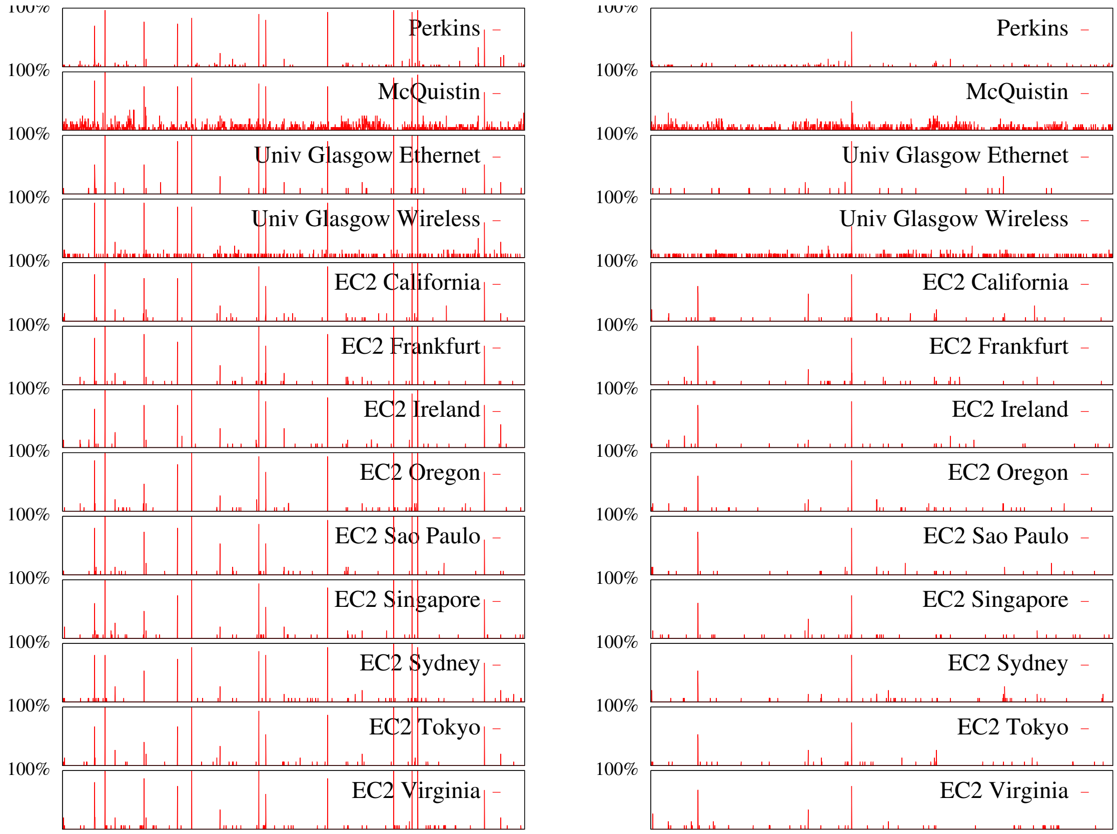
We consider reachability of NTP servers using requests sent in not-ECT marked UDP packets, and in UDP packets sent with an ECT(0) mark. The goal is to characterise differences in server reachability when using ECN, to determine if the presence of an ECT(0) mark on UDP packets makes them more likely to be discarded than not-ECT marked packets. In contrast, Section 4.2 presents a path-based analysis, showing where ECT marks are modified in the network.

Across all traces, an average of 2253 servers from the set of 2500 tested are reachable using not-ECT marked UDP packets. This varies somewhat across traces. The early traces taken in the authors' homes, and the early University of Glasgow wireless traces, show higher reachability than the later traces. These are the traces taken in April/May 2015, whereas those collected in July/August 2015 had lower overall reachability. We believe this is due to servers leaving the NTP pool between the two sets of measurements. We note poor reachability from McQuistin's home, perhaps due to congestion in the access network. We also see more variation in the wireless traces than those collected on wired networks. That some servers are unreachable is not surprising. The NTP pool is operated by volunteers, and offers no service guarantee, so some servers can be expected to be unavailable. Further, UDP is unreliable, and while we retry requests to compensate for packet loss, it can be expected to result in a small number of servers being falsely found unreachable.

Our reachability results are summarised in Figure 2. For each of the 210 traces, we plot a vertical bar in Figure 2a showing the percentage of NTP servers that respond to requests sent in not-ECT marked UDP packets that are also reachable using ECT(0) marked UDP packets. In Figure 2b, we plot the corresponding percentage of servers that respond to requests sent in ECT(0) marked UDP packets that are also reachable using not-ECT marked UDP packets.

The impact of ECT(0) marks on reachability of UDP servers is shown in Figure 2a. Of those servers that respond to not-ECT marked requests, an average of 98.97% also respond to requests sent in ECT(0) marked packets, although this fraction varies somewhat (but is always above 90%) depending on the location from which data is collected. It can be expected that some of these reports are false positives, due to packet loss unrelated to the use of ECN, but some will be caused by middleboxes dropping ECT(0) marked packets. By this measure, and on this dataset, the use of ECT(0) marks generally has a small, but measurable, impact on the reachability of UDP servers (although McQuistin's home network shows that the impact can be larger in some cases).

We also consider the converse, in Figure 2b, where we see that an average of 99.45% of the servers that are reachable with ECT(0)



(a) Reachable by not-ECT but not ECT(0) marked packets (b) Reachable by ECT(0) but not by not-ECT marked packets

Figure 3: Differential reachability of each server using ECT(0) and not-ECT marked packets

marked packets are also reachable using not-ECT marked packets. NTP does not use ECN in its normal operation, so NTP servers configured to drop not-ECT marked UDP packets in this manner, or behind middleboxes with this behaviour, would not be usable for their intended purpose. Accordingly, we believe the unreachable reports for these servers are false, and are due to packet loss that is unrelated to ECN.

To better understand differential reachability when ECN is used, Figure 3a plots, for each server, and from each location, the fraction of traces in which that server is reachable using not-ECT marked packets but not using ECT(0) marked packets. Each vertical bar represents one of the 2500 servers tested. If the server is always reachable with ECT(0) marked packets when reachable with not-ECT marked packets, it will show 0% differential reachability; if it is never reachable using ECT(0) marked packets when reachable using not-ECT packets, it will show 100% differential reachability. Ideally, all servers will be reachable using both ECT(0) and not-ECT marked UDP packets, and hence will show 0% differential reachability.

In practice, the majority of servers have near zero differential reachability. However, a small number of servers (between 9 and 14, depending on the location from which measurements are taken) have differential reachability >50% (these are the tall vertical spikes in Figure 3a). This shows that some servers are generally not reachable with ECT(0) marked UDP packets, but are reachable with not-ECT marked packets, presumably due to firewalls or other middleboxes that drop ECT-marked packets. Visual inspection of Figure 3a shows that it is usually the same set of servers having high differential reachability from every location, suggesting that the ECT(0) marked packets are being dropped near to the destination (we revisit this in Section 4.2).

We note that the differential reachability is high, but not 100%, for some servers. This indicates servers that are usually, but not always, reachable using not-ECT marked packets but not using ECT(0) marked packets. Possible reasons for this might be route changes, causing the middlebox that drops ECT(0) marked packets to be bypassed in some cases, or routers treating the ECN bits as part of the type-of-service field and preferentially dropping such packets. Further study is needed.

Figure 3b shows differential reachability for servers that can be reached using ECT(0) marked packets but not with not-ECT marked packets. As expected, differential reachability is less in this case, with at most 3 servers having differential reachability >50%. Of those, one has high differential reachability from every location tested, while the other two (pool NTP servers run by Phoenix Public Library) seem to be affected in the traces taken from EC2 only. The reasons for the differential reachability of these servers when ECN is not used are unclear.

Overall, we see high reachability of UDP servers with ECT(0) marked packets. While a small number of servers are (sometimes) reachable using not-ECT marked UDP packets but never reachable using ECT(0) marked UDP packets, there are around 4× more servers that are transiently unreachable. Indeed, for the subset of the NTP server pool that we probe, persistent failures due to use of ECN appear to be the least significant cause of reachability problems, behind transient packet loss, and servers that are off-line.

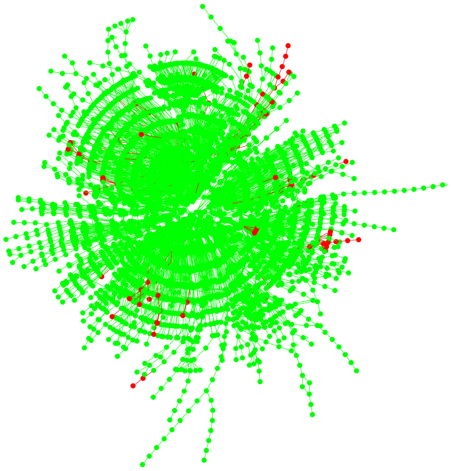


Figure 4: Sample traceroutes, showing hops where ECN is missing

4.2 Are ECN marks stripped from UDP?

The results in Section 4.1 show that use of ECT(0) marks on request packets has only a small impact on reachability of UDP servers. There are two possible reasons why this could be: either the presence of such marks does not significantly affect reachability, or the marks were stripped by a router near the sender and so were not visible to the wider network.

To determine whether the ECT(0) marks were actually traversing the network, we ran traceroutes from each measurement location to each of the NTP servers we identified. The traceroute was configured to send TTL limited ECT(0) marked UDP packets, and we captured returning ICMP responses. We then compared the UDP/IP header encapsulated in the ICMP response with the UDP/IP header sent, to determine whether the ECT(0) mark was present at each hop. This is the same technique used in [1], [2], and [8].

In total, our traceroute data covers 155439 IP level hops in 1400 ASes (subject to the usual limitations of IP to AS mapping accuracy [16]). Representative sample results are presented graphically in Figure 4. The source of the traceroute requests is in the centre of the figure, with the destination servers located at the edges. The path to each server is shown with a dot representing each hop, and lines showing the connections between the hops. IP addresses are omitted, for readability reasons. Hops that return an unmodified ECN field are drawn in green; those where the returned ECN field differs from that sent are shown in red. In all cases, observed changes to the ECN field were to set it to not-ECT, hence we see runs of red in the figure, after the ECT mark has been stripped. We did not see any ECN-CE marks. Traces stop at the point where a traceroute to the server stops; this is generally one hop before the destination.

It is clear that ECT(0) marked packets do traverse the network with their marking intact, in the majority of cases. Of the 155439 hops measured, 154421 pass the ECT(0) mark unmodified, and the mark is stripped at 1143 hops (125 hops only sometimes strip the ECN mark). Regions where ECT marks have been removed, shown in red in Figure 4, are few, widely scattered, and not located near the sender. 59.1% of the locations where ECT(0) marks are stripped, where we were able to determine the AS, were at AS boundaries (again, subject to the limitations of inferring AS number from traceroute IP addresses). This data does not tell us whether marked packets reach their destination with the ECT(0) mark intact, since firewalls that block traceroute might also strip ECN marks, but it does indicate that the marks traverse the wide-area network.

4.3 Reachability using ECN with TCP

We also consider the reachability of the web servers co-located with the NTP pool servers when making HTTP requests using TCP with ECN. Our goals are to determine the fraction of web servers in the pool that successfully negotiate and use ECN, and to compare this to reachability of UDP servers with ECN-marked traffic.

Results are shown in Figure 5. For each trace, the figure shows the number of web servers that respond to requests sent via TCP without using ECN, and the number that successfully negotiate ECN when requested (i.e., the number of servers that respond to an ECN-setup SYN with an ECN-setup SYN-ACK packet). On average, we are able to reach 1334 web servers from the 2500 hosts studied. This is significantly less than the 2253 servers that are reachable on average using UDP. Operators of hosts in the NTP pool are encouraged to run a web server, but it is clear that many do not. As expected, there is little variation in reachability between traces. For those hosts that run web servers, the servers are generally available, and TCP retransmits conceal the impact of packet loss.

Across all traces, the average number of web servers that negotiate ECN support with TCP when requested was 1095 (82.0% of those reachable using TCP). This is considerably lower than the fraction of NTP servers in the pool that were reachable with ECT(0) marked UDP packets, but the results are not directly comparable, since to be recorded as reachable with TCP using ECN, the server needs to actively respond with an ECN-setup SYN-ACK, whereas the UDP reachability test didn't require active participation of the server.[1]

A better comparison is with previous studies of TCP use with ECN. For example, Trammell *at el.* [14] conducted active probes of the Alexa Top million web servers list in 2014 and found 56.17% negotiated ECN when requested. Similar studies by Kühlewind *et al.* [6] found 29.48% would negotiate ECN in 2012, while [1] found 17.2% would negotiate ECN. Langley [7] and Medina *et al.* [9] present earlier data, showing negligible deployment.

Plotting these previous measurements in a time series, along with our new data, gives the result shown in Figure 6. Our results show a significant increase in willingness to negotiate ECN, when compared to the previous measurements, but on a growth curve that looks to be in line with previous results.

Overall results are encouraging, showing successful ECN negotiation with TCP for a high fraction of the servers. We see significantly higher reachability than previous studies, but further work is needed to determine whether the increase is due to measuring against a different set of servers, or whether it is a general increase in TCP ECN reachability.

4.4 UDP and TCP reachability correlation

We compare the servers reachable using unmarked UDP packets but not using ECT(0) marked packets, with the set of servers that do not successfully negotiate the use of ECN with TCP. The goal is to determine if the same servers are unreachable with ECN for both UDP and TCP.

Results are shown in Table 2. There is only weak correlation between servers that are unreachable using UDP with ECT(0), and those that refuse to negotiate ECN with TCP. The majority of servers that cannot be reached using ECN with UDP *can* be reached using ECN with TCP (that is, they will negotiate ECN, then send and receive ECT-marked packet with TCP, but not respond to ECT-marked UDP). This is evidence of middleboxes that discard ECT marked IP packets when the payload is UDP, but not when the payload is TCP.

[1] The ECN-setup SYN is sent in a not-ECT marked packet, so it's also not possible to compare response rates for ECT and not-ECT SYN packets with those for ECT and not-ECT UDP.

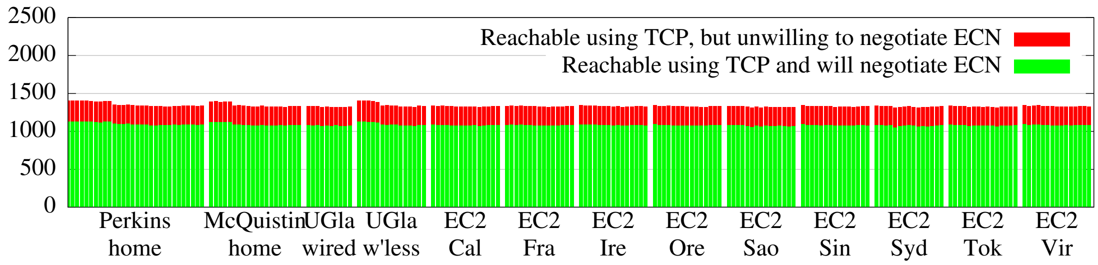


Figure 5: Reachability of web servers using TCP and TCP with ECN, one bar per trace

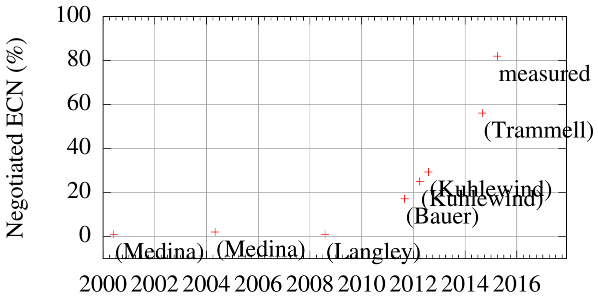


Figure 6: Trends in ECN TCP capability

5. RELATED WORK

Trammell *et al.* [14] probed the Alexa top web servers list, finding 56.17% of IPv4 servers tested in August-September 2014 were willing to negotiate ECN, rising to 65.41% of IPv6 servers. They also studied ECN-dependent connectivity, finding 0.42% of servers showed differences in connectivity when ECN was requested for TCP; this is comparable to our results for UDP.

Kühlewind *et al.* [6] also studied the Alexa top web servers, finding ECN support in 25.16% of servers tested in April 2012, rising to 29.48% in August 2012. Tests conducted against IPv6 hosts show 48.56% successfully negotiating ECN. We find a higher fraction of servers negotiating ECN with TCP, as discussed in Section 4.3. Kühlewind *et al.* also test ECN usability with hosts that negotiate ECN, by sending ECN-CE marked segments and checking whether the returned ACK includes has the ECE flag set, showing approximately 90% usability. We do not perform such a test with TCP, but this result is comparable to our results using UDP in Section 4.2.

Bauer *et al.* [1] performed similar measurements of ECN usability with TCP, testing against both the Alexa server list, and against other University and College web servers, and against mobile sites. The results are broadly comparable to those of Kühlewind *et al.*, although being older, they show less ECN support. Bauer *et al.* also perform traceroute-based probes, similar to those we describe in Section 4.2 although using a larger set of destinations, to determine where ECN marks are modified or stripped in the network. Their results show approximately 82% of traces preserving the ECT bits for the entire path. This is a noticeably lower fraction than we observe, perhaps due to increased awareness of ECN in the operational community.

Langley [7] probed 1,445,303 web servers in 2008, finding approximately 1% negotiated ECN support, and around 0.5% ignored SYN packets sent with ECE and CWR set, but these were not uniformly distributed, with a few providers being responsible for the majority of failures. Medina *et al.* [9] conducted tests of ECN reachability using TCP in 2000 and 2004, with similarly low success.

Location	Avg. unreachable UDP with ECT	Num of those that fail to negotiate ECN w/TCP
Perkins home	8	3
McQuistin home	160	20
U. Glasgow wired	10	2
U. Glasgow w'less	43	4
EC2 California	10	3
EC2 Frankfurt	14	5
EC2 Ireland	11	4
EC2 Oregon	14	2
EC2 Sao Paulo	16	3
EC2 Singapore	10	3
EC2 Syndey	11	5
EC2 Tokyo	13	2
EC2 Virginia	16	3

Table 2: Correlation between UDP and TCP reachability

6. CONCLUSIONS

We present initial results showing how use of ECN affects reachability of UDP servers, testing against 2500 servers from the NTP pool. An average of 98.97% of those reachable with not-ECT marked UDP packets were also reachable using ECT(0) marked packets. The remaining servers were unreachable using ECT(0) marked packets, often persistently so. The use of ECN has a small negative impact on reachability of UDP servers (Section 4.1). Further measurements show that ECT(0) marks successfully traverse most (~98%) reachable network hops unmodified, but have the ECT mark set back to not-ECT in the remaining cases (Section 4.2).

We test reachability of the same servers using TCP with ECN, finding 82.0% of those reachable with TCP will negotiate ECN support (Section 4.3). This is higher than previous studies, and indicates that ECN is becoming usable with TCP. Comparison of TCP and UDP reachability when using ECN (Section 4.4) shows poor correlation between servers unreachable using ECT(0) marked UDP and servers that refuse to negotiate ECN with TCP. Some paths allow ECT(0) packets when the payload is TCP, but not for UDP.

While our dataset is comparatively small, and our measurements were taken from a small number of locations, the servers we probe are located at a wide range of locations around the world, and in many different network environments. Ongoing studies, to verify our results in more environments, would be welcome. To the extent that they are representative, though, our results show that marking UDP packets with ECT(0) will not, in general, harm reachability. Whether the use of ECN with UDP offers any benefit has not been determined, but it seems to cause no significant harm.

Acknowledgements: We thank the anonymous reviewers, and our shepherd, Matthew Luckie, for their feedback.

7. REFERENCES

[1] S. Bauer, R. Beverly, and A. Berger. Measuring the state of ECN readiness in servers, clients, and routers. In *Proceedings of the Internet Measurement Conference*, Berlin, Germany, November 2011. ACM.

[2] G. Detal, B. Hesmans, O. Bonaventure, Y. Vanaubel, and B. Donnet. Revealing middlebox interference with tracebox. In *Proceedings of the Internet Measurement Conference*, Barcelona, Spain, October 2013. ACM.

[3] J. Hildebrand and B. Trammell. Substrate protocol for user datagrams (SPUD) prototype. Internet Engineering Task Force, March 2015. Work in progress (draft-hildebrand-spud-prototype).

[4] Internet Architecture Board. Workshop on stack evolution in a middlebox Internet, January 2015. https://www.iab.org/activities/workshops/semi/.

[5] C. Jennings, T. Hardie, and M. Westerlund. Real-time communications for the web. *IEEE Communications Magazine*, 51(4):20–26, April 2013.

[6] M. Kühlewind, S. Neuner, and B. Trammell. On the state on ECN and TCP options on the Internet. In *Proceedings of the Passive and Active Measurement Conference*, Hong Kong, China, March 2013. Springer.

[7] A. Langley. Probing the viability of TCP extensions. https://www.imperialviolet.org/binary/ecntest.pdf, September 2008.

[8] D. Malone and M. Luckie. Analysis of ICMP quotations. In *Proceedings of the Passive and Active Measurement Conference*, Louvain-la-neuve, Belgium, April 2007.

[9] A. Medina, M. Allman, and S. Floyd. Measuring the evolution of transport protocols in the Internet. *ACM Computer Communication Review*, 35(2):37–52, April 2005.

[10] K. K. Ramakrishnan, S. Floyd, and D. L. Black. The addition of ECN to IP. Internet Engineering Task Force, September 2001. RFC 3168.

[11] J. Rosenberg et al. SIP: Session initiation protocol. Internet Engineering Task Force, June 2002. RFC 3261.

[12] J. Roskind. Experimenting with QUIC, June 2013. https://blog.chromium.org/2013/06/experimenting-with-quic.html.

[13] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: A transport protocol for real-time applications. Internet Engineering Task Force, July 2003. RFC 3550.

[14] B. Trammell, M. Kühlewind, D. Boppart, I. Learmonth, G. Fairhurst, and R. Scheffenegger. Enabling Internet-wide deployment of explicit congestion notification. In *Proceedings of the Passive and Active Measurement Conference*, New York, NY, USA, March 2015.

[15] M. Westerlund, I. Johansson, C. S. Perkins, P. O'Hanlon, and K. Carlberg. ECN for RTP over UDP. Internet Engineering Task Force, August 2012. RFC 6679.

[16] Y. Zhang, R. Oliveira, H. Zhang, and L. Zhang. Quantifying the pitfalls of traceroute in AS connectivity inference. In *Proceedings of the Passive and Active Measurement Conference*, Zurich, Switzerland, April 2010.

[17] X. Zhu et al. NADA: A unified congestion control scheme for real-time media. Internet Engineering Task Force, March 2015. Work in progress (draft-ietf-rmcat-nada).