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 measurements to determine if ECN is usable with UDP flows 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 increasing use of UDP as a substrate on which new transport protocols can be deployed. We show that UDP-based servers can be reached using ECT(0) marked packets with high probability, and that most 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 around 80% can successfully negotiate and use ECN in that case. Our findings suggest that ECN is broadly usable with UDP traffic, and that support for use of ECN with TCP has increased.

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 [8] 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.

ECN relies on two related mechanisms: IP header flags to indicate support for ECN and to mark packets when congestion occurs, and transport layer mechanisms to inform the sender when a marked packet is received, allowing the sender to respond to the congestion. Support for ECN with TCP has long been defined [8]. More recently, there has been interest in using ECN with transport protocols that layer above UDP.

ECN can be used with UDP to support interactive multimedia applications, such as those using the WebRTC framework [4]. WebRTC uses RTP [11] for media transport, and one of us recently developed RTP extensions for ECN feedback [12]. Congestion control algorithms for interactive video using RTP are under development in IETF, and one of the candidates, NADA [13], makes extensive use of ECN. This is desirable for interactive video, since ECN support within 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. ECN can significantly improve the user experience.

Other cases where ECN is helpful to UDP-based transports include congestion feedback for tunnelled pseudo-wire traffic, and to support UDP use as a substrate for new transport protocol development. Examples of the latter include QUIC [10], SPUD [2], and various proposals for IP stack evolution discussed at the recent IAB workshop on Stack Evolution in a Middlebox Internet [3]. It is clear that UDP-based transport protocols could benefit from the use of ECN, provided it is both usable and deployable in the current Internet.

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 when using ECN, we show that marking UDP packets as ECN capable has a small, but non-zero, impact on reachability of UDP servers, with ~0.4% of servers that are reachable using unmarked UDP packets being unreachable when using UDP packets marked as ECN capable. This is approximately 4× fewer than the fraction of servers that are transiently unreachable due to packet loss. For comparison, we measure TCP reachability to the same set of servers, noting that approximately 80% of the servers that are reachable using TCP will successfully negotiate ECN. This represents a significant increase in ECN support for TCP, compared with previous studies.

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, our measurements show that ECN has the potential to be widely used with TCP, demonstrating around 2.5× more support than the last study of which we are aware.

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 is a best effort network, relying 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 as a response, completing the feedback loop and ensuring stability. 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. The sender reacts to this indication as it would to loss.

The ECN standard [8] takes two bits from the IP header and sets them to indicate whether the 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 regarded as 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 experienced on the path. Receipt of ECN-CE marked packets triggers the transport on the receiver to send an indication to the sender, closing the feedback loop.

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 [11], for which appropriate ECN feedback is defined in [12]. The use of ECN with RTP is negotiated using a non-RTP signalling channel, such as SIP [9] or WebRTC [4], and both endpoints need to agree to its use before data packets are 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. Of interest is whether it is necessary to also probe the channel to determine if the use of ECT-marked UDP packets affects reachability, or whether marked packets can be assumed to reach their destination with the same probability as any other UDP packet.

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

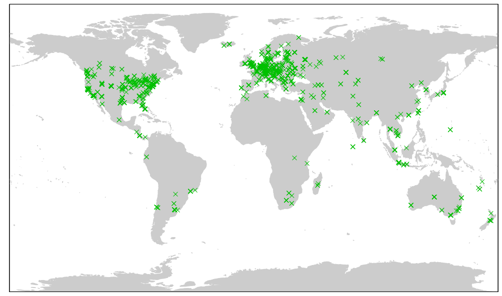

Figure 1: Geographic locations of NTP pool servers

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

are also reachable using TCP. A set of servers that meet these constraints are the network time protocol (NTP) pool servers.

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 (or one of its country- or region-specific sub-domains) when they start-up. The pool operates a 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.

The locations of these NTP servers were found using the MaxMind GeoLite2 City database, as of 25 April 2015, and are shown in Figure 1. Their geographic distribution is sum-

Trace	Reachable by UDP	Reachable by UDP with ECT	Reachable by UDP, not UDP with ECT	Reachable by UDP wth ECT, not UDP
P1	2404	2390	19	5
P2	2404	2392	15	3
P3	2406	2394	13	1
P4	2404	2391	15	2
P5	2395	2385	13	3
P6	2391	2380	14	3
P7	2386	2376	14	4
P8	2380	2369	15	4
P9	2385	2373	13	1
P10	2390	2375	17	2
M1	2302	2282	68	48
M2	2356	2353	32	29
M3	2361	2352	28	19
M4	2346	2330	39	23
M5	2368	2361	28	21
W1	2373	2354	71	52
W2	2363	2338	68	43
W3	2362	2352	54	44
W4	2371	2353	51	33
W5	2396	2385	16	5

Table 2: Reachability of NTP servers using UDP and UDP with ECT(0) marks, broken down per trace

marised 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 20 traces, where each trace tests both protocols, with and without the use of ECN, against each of the 2500 servers. The traces are split into three traces sets: the M and P trace sets were taken from the homes of the two authors, whereas the W trace set was taken from their workplace at the University of Glasgow. Traces were collected in late April and early May 2015. The P trace set had the lowest UDP packet loss rate, as discussed in Section 4.1, and hence more data was collected from that location for this initial study. 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, after a timeout starting at one second and doubling with 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 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.

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 these 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 complete data set comprises 20 such traces.

4. RESULTS

We present two main sets of results, measuring reachability using UDP packets with ECN in Section 4.1, and using TCP with ECN in Section 4.3. These are supplemented with a discussion of whether and where ECN marks are stripped from UDP packets in Section 4.2, and a comparison of reachability between the two protocols in Section 4.4.

4.1 Reachability using ECN with UDP

We consider reachability using requests sent using not-ECT marked UDP packets, and using UDP packets sent with an ECT(0) mark. The goal is to characterise differences in reachability, to determine if the presence of an ECT(0) mark on UDP packets makes them more likely to be dropped by middleboxes than not-ECT marked packets.

Our results are summarised in Table 2. For each trace, we record the number of servers reachable using not-ECT marked UDP, the number that are reachable using UDP with an ECT(0) mark, the number that are reachable using not-ECT marked UDP but not using UDP with an ECT(0) mark, and finally the number reachable using UDP with an ECT(0) mark but not using not-ECT marked UDP.

An average of 2377 servers from the set of 2500 tested are reachable using not-ECT marked UDP packets. This varies somewhat across traces, with the P set of traces having generally higher reachability than the M and W traces. 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. Furthermore, UDP is unreliable, and while we compensate for packet loss as described below, it can be expected to result in a small number of servers being falsely found unreachable.

We now consider the impact of ECN on reachability of

UDP servers. The number of NTP pool servers reachable using UDP with ECT(0) marked requests was 99.5% of the number reachable using not-ECT marked UDP in the P traces. Results for the other traces were similar: 99.5% for the M traces, and 99.3% for the W traces. By this measure, and on this dataset, the use of ECT(0) marks has a small, but measurable, impact on the reachability of UDP servers.

A small number of servers show as reachable via UDP packets marked with ECT(0), but not with not-ECT marked UDP packets. This is less common in trace set P than in trace sets M and W, but occurs in all sets. NTP does not use ECN in its normal operation, so NTP servers configured 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 in the network that is unrelated to ECN. This suggests there is a higher underlying packet loss rate in the networks used to collect the M and W traces, than in that used to collect the P trace set.

A larger fraction of the servers are reachable via not-ECT marked UDP, but not UDP with an ECT(0) mark. Again, the numbers are lower for the P set of traces than for the M and W sets, but this occurs for all sets. As discussed previously, 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 that drop ECT(0) marked packets.

The number of reachable servers within each category is broadly consistent across traces in each trace set, but there is noticeably higher reachability (equivalently, lower numbers of servers reachable with one UDP variant than the other) in the P traces than in the M and W traces. As we have noted, we believe this is largely due to differences in the underlying packet loss rate for the different networks.

To determine which servers were unreachable due to packet loss unrelated to ECN, and when the loss is ECN related, we combine the data from all traces. The goal is to determine whether servers that are unreachable in one trace are reachable in another. The results are shown in Table 3.

The number of unreachable servers decreases when results are aggregated, showing the majority are, at least sometimes, reachable. In particular, there are no servers that are reachable via UDP with ECT(0) marks, but not with not-ECT marked UDP. We observe a core of NTP servers that were never reachable using UDP, and appear to be off line for the duration of our measurements. Finally, there are a small number of servers (~0.4% of the total) that are reachable using UDP in at least some of each set of traces, but that never respond to NTP queries sent using UDP with ECT(0) marks. Our assumption is that these servers are behind middleboxes that discard UDP packets sent with an ECT(0) mark.

Review of the IP addresses of the servers that are never reachable with ECT(0) marked packets shows that the unreachable servers in the M and W trace sets are a superset of those in the P trace set. There are two ECT(0) unreachable servers in the M trace set that are reachable in the other trace sets, and similarly one ECT(0) unreachable server in the W trace set that is reachable in the others. This suggests that there are some locations within the network that discard ECT(0) marked packets, leading to differential reachability. We revisit this point in Section 4.2.

Set	Reachable via UDP, not UDP with ECT(0)	Reachable via UDP with ECT(0), not UDP	Not reachable
P	4	0	72
M	9	0	86
W	8	0	67

Table 3: Overall number of unreachable NTP servers

Overall, we see extremely high reachability for UDP-based servers with ECN. While a small number of servers are (sometimes) reachable using not-ECT marked UDP packets, but that are 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 the use of ECN appear to be the least significant cause of reachability problems, behind transient packet loss, and servers that are off-line.

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 traceroute from the server for trace set P (since it had least non-ECN related UDP packet loss) 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.

The results are presented graphically in Figure 2. 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 ECN field is changes are shown in red. In all cases, observed changes to the ECN field were to set it to not-ECT; 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 sent from this location do traverse the network with their marking intact, in the

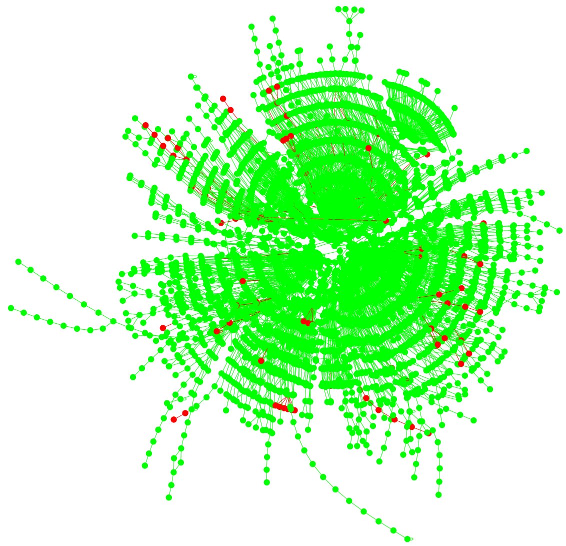

Figure 2: Server traceroutes, showing change in encapsulated ECN field

majority of cases. Of the 8611 hops measured, 8449 pass the ECT(0) mark unmodified, and the mark is removed by 162 hops. Locations where ECT marks are stripped, shown in red in Figure 2, are few, widely scattered, and not located near the sender. 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 colocated 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.

Our reachability results are shown in Table 4. For each trace, the table shows the number of web servers that respond to requests sent via TCP without using ECN, and the number that are reachable and 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 1396 web servers from the 2500 hosts studied. This is significantly less than the 2377 servers that are reachable 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 the traces, the average number of web servers that negotiated ECN support with TCP when requested was 1130 (80.9% of those reachable using TCP without ECN). This is considerably lower than the fraction of NTP servers

Trace	Reachable by TCP	Reachable by TCP with ECN
P1	1404	1137
P2	1403	1133
P3	1404	1135
P4	1402	1133
P5	1401	1133
P6	1398	1130
P7	1393	1128
P8	1393	1119
P9	1395	1129
P10	1397	1130
M1	1391	1132
M2	1392	1131
M3	1386	1130
M4	1388	1129
M5	1390	1132
W1	1406	1135
W2	1401	1132
W3	1401	1124
W4	1395	1124
W5	1385	1126

Table 4: Reachability using TCP and TCP with ECN

in the same 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.

A better comparison is with previous studies of TCP use with ECN. For example, Kühlewind *et al.* [5] conducted active probes of the Alexa Top 100,000 web servers list and found 29.48% negotiated ECN when requested. Bauer *et al.* [1] studied a similar set of servers from the Alexa list some years earlier, and found 17.2% would negotiate ECN. Langley [6] and Medina *et al.* [7] 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 3. Our results show a significant increase in willingness to negotiate ECN, when compared to the previous measurements, but on a growth curve that does not look exceptional. We also note that the servers we measure, being a low traffic service operated by volunteers, will likely have different characteristics and configuration to servers in the Alexa list of most popular web servers worldwide, so some variation in reachability is to be expected.

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, and 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.

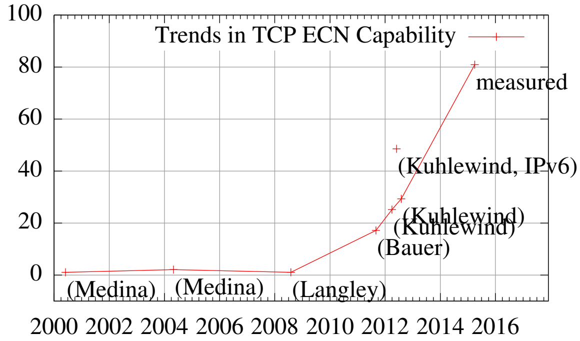

Figure 3: Trends in ECN TCP capability

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 5. There is strong, but not perfect correlation. A small number of servers that never respond to ECT(0) marked UDP requests, successfully negotiate ECN with TCP. Review of the tcpdump logs shows these servers do respond to ECT(0) marked TCP data segments. This is evidence of middleboxes that discard ECT marked IP packets when the payload is UDP, but not when the payload is TCP.

5. RELATED WORK

Kühlewind *et al.* [5] conducted probes of the Alexa top web servers list, finding the fraction that will negotiate ECN with TCP. They find 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 with TCP. We show 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 have negotiated ECN for a TCP flow, 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. Finally, they conduct passive measurements on the SWITCH network backbone, to observe ECN deployment in the wild, finding little usage.

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

Set	Number unreachable via UDP with ECT(0)	Number of those that fail to negotiate ECN with TCP
P	4	3
M	9	7
W	8	6

Table 5: Correlation between UDP and TCP reachability

for the entire path. This is a noticeably lower fraction than we observe, perhaps growing increasing awareness of ECN in the operational community over time.

Langley [6] probed 1,445,303 web servers in August 2008, showing that 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.* [7] conducted tests of ECN reachability using TCP in 2000 and 2004, with similarly low success.

6. CONCLUSIONS

We presented initial measurement results showing how use of ECN affects reachability of UDP servers, testing against 2500 servers from the NTP pool. These show that ECT(0) marked packets are deterministically dropped on routes to a small number (~0.4%) of servers, but that in the remainder of cases the use of ECN doesn't affect reachability of UDP services (Section 4.1). The number of servers that were persistently unreachable due to use of ECN was around 4× less than the number transiently unreachable due to packet loss unrelated to ECN. Further measurements show that ECT(0) marks successfully traverse the majority (~98%) of reachable network hops unmodified, but have the ECT mark set back to not-ECT in the remaining cases (Section 4.2).

We also measure reachability of the same servers using TCP with ECN, finding 80.9% of those reachable with TCP will successfully negotiate ECN support (Section 4.3). This is higher than previous studies, and indicates that ECN is becoming more usable with TCP. Comparison of TCP and UDP reachability when using ECN (Section 4.4) shows strong, but not perfect, correlation between servers that are unreachable using ECT(0) marked UDP and servers that refuse to negotiate ECN with TCP. Some paths will allow ECT(0) packets when the payload is TCP, but not when it is 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.

7. REFERENCES

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

[2] J. Hildebrand and B. Trammell. Substrate protocol for user datagrams (SPUD) prototype. IETF, March 2015. Work in progress (draft-hildebrand-spud-prototype).

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

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

[5] M. Kühlewind, S. Neuner, and B. Trammell. On the state on ECN and TCP options on the Internet. In *Proc. PAM*, Hong Kong, March 2013.

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

[7] A. Medina, M. Allman, and S. Floyd. Measuring the evolution of transport protocols in the internet. *ACM CCR*, 35(2):37–52, April 2005.

[8] K. K. Ramakrishnan, S. Floyd, and D. L. Black. The addition of ECN to IP. IETF, September 2001. RFC 3168.

[9] J. Rosenberg et al. SIP: Session initiation protocol. IETF, June 2002. RFC 3261.

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

[11] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: A transport protocol for real-time applications. IETF, July 2003. RFC 3550.

[12] M. Westerlund, I. Johansson, C. S. Perkins, P. O'Hanlon, and K. Carlberg. ECN for RTP over UDP. IETF, August 2012. RFC 6679.

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