Evaluating the Benefits: Quantifying the Effects of TCP Options, QUIC, and CDNs on Throughput

Simon Bauer
Technical University of Munich
bauer@net.in.tum.de

Patrick Sattler
Technical University of Munich
sattler@net.in.tum.de

Johannes Zirngibl
Technical University of Munich
zirngibl@net.in.tum.de

Christoph Schwarzenberg
Technical University of Munich
schwarzenberg@net.in.tum.de

Georg Carle
Technical University of Munich
carle@net.in.tum.de

ABSTRACT

To keep up with increasing demands on quality of experience, assessing and understanding the performance of network connections is crucial for web service providers. While different measures, like TCP options, alternative transport layer protocols like QUIC, or the hosting of services in CDNs, are expected to improve connection performance, no studies are quantifying such impacts on connections on the Internet.

This paper introduces an active Internet measurement approach to assess the impacts of mentioned measures on connection performance. We conduct downloads from public web servers considering different vantage points, extract performance indicators like throughput, RTT, and retransmission rate, and survey speed-ups due to TCP option usage. Further, we compare the performance of QUIC-based downloads to TCP-based downloads considering different option configurations.

Next to significant throughput improvements due to TCP option usage, in particular TCP window scaling, and QUIC, our study shows significantly increased performance for connections to domains hosted by different giant CDNs.

CCS CONCEPTS

• **Networks → Public Internet**; **Transport protocols**.

KEYWORDS

TCP options, QUIC, CDNs, performance measurements, Internet measurements

ACM Reference Format:
Simon Bauer, Patrick Sattler, Johannes Zirngibl, Christoph Schwarzenberg, and Georg Carle. 2023. Evaluating the Benefits: Quantifying the Effects of TCP Options, QUIC, and CDNs on Throughput. In *Applied Networking Research Workshop (ANRW '23), July 24, 2023, San Francisco, CA, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3606464.3606474

ANRW '23, July 24, 2023, San Francisco, CA, USA

ACM ISBN 979-8-4007-0274-7/23/07…$15.00
https://doi.org/10.1145/3606464.3606474

1 INTRODUCTION

Due to its impact on user satisfaction, understanding the performance of connections and the impact of potential performance improvements is crucial for service and infrastructure providers. The same applies from a research perspective to assess the effectiveness of arising or widely deployed measures to improve connection performance.

The transmission control protocol (TCP), responsible for the majority of Internet traffic [1], was extended for several options proposed to improve the performance shortcomings of its original design. Further, QUIC and HTTP3 represent an arising alternative to the commonly used TCP/HTTPS stack [2]. This development also motivates a closer look at performance differences between QUIC and TCP connections targeting the same resources in productive deployments, i.e., on the Internet. Next to protocol usage and their configuration, the hosting of service infrastructure is crucial to ensure availability and performance. This led to the trend towards a more centralized Internet, i.e., more services being hosted in large-scale content delivery networks (CDNs) [3–6].

However, while there are publications surveying deployments and usage of different TCP options [7–9], comparing the performance of TCP and QUIC connections in controlled test environments [10, 11], or focusing on optimizations of both protocol stacks [12], there are no insights on the impact of TCP option usage, QUIC usage, or CDN hosting on connection performance on the Internet.

This paper assesses the impacts of the named measures on connection performance by conducting active measurements with public Internet servers. Thereby, we exploit the capability of active Internet measurements to determine client configurations and target selection.

Our contributions in this work are:

(i) We introduce an active measurement approach for public Internet web servers covering crawling of suitable measurement targets, conducting downloads with different client configurations, and analyzing the performance of connections by extracting different performance indicators.

(ii) We apply the introduced approach to measurements from different vantage points on a set of publicly available web servers chosen from Internet top lists and discuss corresponding measurement results in this paper.

(iii) We publish the implemented measurement pipeline.

2 BACKGROUND

Performance-related TCP Options. In this paper, we consider three options addressing the performance of TCP connections: window scaling (WS), selective acknowledgments (SACK), and explicit congestion notifications (ECN). TCP window scaling [13] is proposed to exceed limitations of bytes in flight implied by the length of the `window` field in the TCP header from 65 kB up to 1 GB. TCP WS enables the exchange of a factor during connection establishment to shift all following values of the `window` field.

Selected acknowledgments [14] are purposed to avoid unnecessary retransmissions by specifying lost packets. This is achieved by exchanging ranges of sequence numbers of successfully received packets. Today, WS and SACK are enabled on all major operating systems like Linux, MacOS, or Windows [8].

Explicit congestion notifications (ECN) [15] are a measure to avoid overload on the network and, accordingly, lost packets by enabling routers to signal congestion actively. As routers do not access Layer 4 headers, ECN relies on two bits each in a packet's IP and TCP header. Permutations of such flags are then used to signal ECN support, detected congestion, and reaction to observed congestion. In addition to the endpoints, all routers of a network path have to support ECN. Typical operating systems support the usage of ECN for at least incoming connections [9].

Note that we do not consider TCP Fast Open (TFO) [16] for this study. TFO is particularly effective if a client requests several sources from a server, resulting in several TCP handshakes implying overhead. This use case does not match our approach to explicitly download single files from a target, as described in Section 4.

QUIC. QUIC is a transport layer protocol specified in 2021 [17–19], providing properties like reliable data transmission, connection migration, and encryption while relying only on UDP packet sequences. QUIC implements selective acknowledgments by acknowledging ranges of packet numbers indicating lost packets. In contrast to TCP, which limits selective acknowledgments to a maximum of three ranges of sequence numbers, QUIC supports up to 256 ranges. Further, QUIC supports ECN usage. Upper bounds for receiver window sizes differ between QUIC implementations.

3 RELATED WORK

TCP options and their deployment are frequently addressed topics. Studies of TCP option deployments conducted in the early 2000s observed the evolution of option deployment, starting from only small adoption of servers to TCP options [20–22]. A study conducted in 2013 by Kühlewind et al. [7] reports widespread deployment of WS and SACK and observes a slower spreading of ECN usage. Such observation is confirmed by Murray et al. [8] in 2017, who only observed small usage of ECN in captured Internet traffic from a university network. More recent studies observe that ECN is used by the majority of domains listed in the Alexa Top 1 M list [23], respectively, in passively captured university network traffic [9]. The interference of middleboxes on TCP options is surveyed by Honda et al. [24]. Edeline and Donnet [25] survey the impact of TCP option usage in controlled test environments showing the beneficial effects of TCP options.

According to W3Techs [26] QUIC accounted for 8% of the total global Internet traffic in 2022. Shreedhar et al. [27] compare QUIC to the TCP/TLS stack and observe significantly smaller connection duration for web workloads on the Internet. However, TCP option usage is not considered by the study. Further publications show that QUIC outperforms TCP in different controlled test environments [10, 11, 28].

Additional studies survey quality of experience (QoE) metrics of different web applications based on passive data sets [29–31]. In contrast, our work analyzes transport layer performance based on active measurements. The reproduction of realistic web applications and web pages for performance measurements was studied by Jun et al. [32] and Zilberman et al. [33]

Considering the above state of the art, there are only limited insights into the impact of TCP option usage on the performance of Internet connections, while TCP options are commonly deployed. The same applies to the implications of QUIC usage and the impacts of CDN hosting on connection performance.

4 APPROACH

For our study, we download files provided by public web servers taken from Internet top lists with varying TCP options and QUIC. This section describes the different steps of our active measurement approach, like determining and selecting suitable measurement targets and conducting downloads with controlled client configurations. Considered performance indicators and other extracted metrics are introduced in Section 5.

4.1 Determining Measurement Targets

Conducting active measurements with public and uncontrolled targets on the Internet requires crawling domains for suitable files for download. We refer to a suitable file if it satisfies a specific minimum file size, purposed to provide comparable results between different domains. For our study, we choose a minimum file size of 1 MB. While this is a relatively large file size considering the distribution of flow sizes observed in passive data sets [1], the same study emphasizes the relevance of connections with such size regarding the totally observed bytes.

Based on suitable target domains and corresponding files, we compose a target set considering six different groups regarding CDN hosting. Firstly we consider domains hosted by different giant CDNs, i.e., Cloudflare, Akamai, Amazon, Google, and Microsoft (200 domains for each). Secondly, a sixth target group consisting of targets not hosted by the listed organizations (1000 domains). We map domains to the selected hosters based on the used IP address, its mapping to the announcing autonomous system (AS) based on BGP dumps from a Route Views [34] collector, and a mapping of ASes to their respective organizations based on the work from Arturi et al. [35]. We ensure that all selected domains support the three considered TCP options.

4.2 Conducting Downloads

Next, we initiate downloads of the crawled files for each domain in the composed target set. Before each download, we freshly resolve the target domain to ensure adaption to DNS-based load balancing. We consider different permutations of TCP option usage for downloads and conduct one download for each configuration for a domain sequentially. Afterward, we continue with downloads from the following domain in the target list.

We examine a specific collection of TCP option configurations, which include: (i) a baseline configuration (BL) that does not utilize any options, (ii) configurations supporting only one of the considered options (ECN, SACK, WS14), (iii) a configuration that enables all options (ALL14), and (iv) two configurations that combine window scaling with SACK and ECN, respectively. We use the maximum window scaling factor of 14 for all listed configurations. To survey the impact of the scaling factor, we consider a configuration supporting all options and a scaling factor of of 7 (ALL7). To allow a fair comparison between TCP and QUIC, we only conduct TCP downloads via HTTPS, implying that TCP and QUIC connections provide encrypted data transmission.

By design, this measurement approach is limited to control configuration and conditions at the client. This implies that the server and its characteristics, like operating systems, server implementations, or used congestion control algorithms, are not known. The same applies to load conditions on the Internet paths and at the target server. We conduct downloads with different configurations back to back for one domain, referred to as one measurement run. This procedure ensures that conditions in the network and on the server side are as similar as possible for all downloads of one run, e.g., regarding daytime patterns of service usage and corresponding load. Further, measured performance indicators are compared within one measurement run, for instance, to calculate speed-ups by a configuration.

4.3 Vantage Points

As connection performance also depends on the location of the vantage point (VP), considering the distance to target servers and last-mile network conditions, we use three different vantage points for our measurements: First, a physical server located in a campus data center in Munich (MUC) and, second, two virtual machines hosted by the cloud provider DigitalOcean in data centers located in San Francisco (SFO) and Singapore (SGP). The physical server hosted in Munich is connected with a 1 $\frac{\text{Gb}}{\text{s}}$ up- and downlink to the German science network (DFN) that connects to the Internet via a major Tier 1 provider. The measurement host is equipped with an Intel Xeon E5-2630 CPU providing six physical cores at a clock frequency of 2.6 GHz, 32 GB memory, and a Broadcom NetXtreme BCM5719 Gigabit NIC. The virtual machines hosted in SFO and SGP are equipped with two virtual CPU cores and 4 GB memory.

4.4 Ethical Considerations

Active measurements on public infrastructure like the Internet require responsible measurement practices. We followed a set of ethical measures, i.e., informed consent [36] and community best practices [37] during all our scans. Our measurement hosts' IP addresses can be identified via reverse DNS or WHOIS information, while the measurement host operates an explanatory website. We maintain an abuse contact email and react quickly to all requests, including the option to exclude a domain or IP range from further measurements. We use a custom HTTP user agent to be identifiable as a research group and follow crawling instructions in the robots.txt according to the Robots Exclusion protocol [38].

5 IMPLEMENTATION

This section describes the implementation of the different measurement pipeline components. The implemented pipeline is publicly available [39].

5.1 Crawling and Conducting Downloads

Based on a set of domains, crawling aims to identify web servers providing suitable files, as described in Section 4. In order to determine files for downloads, the crawler recursively follows links found on a crawled website if links explicitly point to the same domain and can be reached by the same IP address as the initially crawled website. The crawling component sends HTTP HEAD requests to extract the optional HTTP Content-Length field [40] with the Python crawling library *Scrapy* to determine the size of a crawled file. First tests showed that many targets do not provide Content-Length information. Accordingly, we implemented a fallback by starting downloads of files and stopping them if the minimum file size was successfully downloaded.

Crawling targets results in a list of domains and corresponding files suitable for our downloads. The download component iterates over the list of determined files and conducts a download with each specified TCP configuration, respectively QUIC. Before downloads are established, the downloader resolves the target domain's IP address to ensure an up-to-date resolution. Afterward, an HTTPS GET request is sent to the freshly resolved target IP with the python *HTTP Requests* library. We use the *ForcedIPHTTPS adapter* [41] to enable the use of specific IP addresses while establishing a TLS/SSL connection. Download traffic gets captured with *tcpdump*.

The downloader relies on the corresponding settings of the Linux kernel to configure TCP option usage. In particular, the downloader sets flags indicating ECN, SACK, and WS usage and sets the kernel's TCP receive memory size to enforce the use of a specific window scaling factor. To conduct QUIC downloads, we rely on the QUIC implementations *aioquic* [42] and *quiche* [43]. Considering different QUIC implementations is motivated by a recently conducted study comparing the performance of QUIC implementations in controlled test environments [44]. According measurement results show that *quiche* outperforms *aioquic* in high bandwidth scenarios. This observation motivates to survey whether such performance differences are also represented in downloads conducted on the Internet. Integrating additional QUIC implementations is a considered extension of our measurement pipeline.

5.2 Traffic Analysis

For our study, we primarily examine the throughput of connections. We calculate average throughput, in the following referred to as mean throughput, as the fraction of transmitted data and the duration of a connection. The amount of transferred data is determined by the sum of packet sizes of a connection, as specified by the *total length* field in IP packet headers.

In addition, the traffic analysis component also extracts performance indicators like mean round trip time (RTT), total retransmission rate (RR), or goodput to further survey the performance characteristics of analyzed connections. We calculate the RTT based on the TCP timestamp option, which enables matching tuples of

Table 1: Number of domains resulting in successful downloads.

Run	Total	Akamai	Amazon	Cloudflare	Google	Microsoft	Others
TCP							
MUC	1692	178	162	159	159	181	853
SFO	1558	173	156	65	154	180	830
SGP	1541	169	157	65	154	178	818
QUIC							
MUC_Q	511	3	15	289	2	0	202
SFO_Q	506	3	14	285	2	0	202
SGP_Q	495	3	13	276	2	0	201

corresponding packets based on *TSval* and *TSecr* values. The difference of corresponding packet timestamps then determines an RTT sample. We determine packets as retransmissions according to the Wireshark documentation considering retransmissions, fast retransmissions, and spurious retransmissions [45]. The retransmission rate is then calculated as the fraction of data transported by retransmitted packets and the total number of observed data. Goodput is calculated as the fraction of the sum of packet sizes without considering retransmissions and the duration of a connection. Further, we extract different IP and TCP header fields to survey the effective use of TCP options like ECN echo and CWR flags or SACK blocks from the TCP header. For QUIC connections, we only consider mean throughput as a performance indicator, calculated in the same manner as for TCP.

6 INTERNET MEASUREMENTS

This section evaluates performance indicators extracted during conducted downloads. In addition to connection performance, this section surveys the option deployment across domains in the Alexa Top 1M list.

6.1 Option Deployment

To provide a recent view on option deployment on the Internet, we conduct active measurements to all domains included in the Alexa Top 1M list (the selected list contained 1M domains). Measurements are conducted by establishing TCP handshakes to the index page of each domain. After establishing the handshake, we immediately terminate TCP connections, as the handshake is sufficient to extract supported options. 5.3 % of domains do not support a single option while 81.0 % support all three considered options. ECN is supported by 85.8 %, SACK by 91.4 % and WS by 91.1 % of the domains.

6.2 Targets for Performance Measurements

To compose our target set, we crawl the top 100K entries of the Alexa Top 1M list. Crawling results in over 22K measurement targets providing a file of at least 1 MB. We select 2000 domains according to the organization maintaining the AS number of a domain, as described in Section 4, referred to as the TCP target set. We observe that not all downloads from successfully crawled domains succeed, for instance, because crawled files are no longer available. In addition, we observe that around a hundred domains hosted by Cloudflare do not result in successful downloads for the vantage points in SFO and SGP, while downloads succeed from the vantage point in MUC. This observation indicates that Cloudflare blocked some of our download attempts, e.g., through human verification for selected IP ranges [46].

We find only negligible shares of domains resulting in successful QUIC downloads in the TCP target set. Therefore, we compose a second target set, referred to as the QUIC target set. As the Alexa Top 1M list was retired in February 2023, the QUIC target set comprises domains taken from the top 100K entries of Google's CrUX dataset [47]. We determine domains supporting QUIC with the *QScanner* introduced by Zirngibl et al. [48]. Based on the list of domains supporting QUIC, we choose targets providing a suitable file and supporting all considered TCP options. Finally, we merge QUIC-supporting targets from the Alexa-based TCP target set with the domains taken from the CrUX dataset. This procedure results in 558 suitable measurement targets. In the future, scanning for QUIC targets might be replaced by analyzing HTTPS DNS resource records which i.a., provide information regarding a domain supporting QUIC. However, Zirngibl et al. have shown that the record is currently used mainly by Cloudflare [49].

We run three measurement iterations for the TCP target set, while one iteration consists of one measurement run per domain. For the significantly smaller QUIC target set, we conduct ten measurement iterations. Note that measurements based on the QUIC target set only consider downloads with *aioquic*, *quiche*, TCP-BL, and TCP-ALL14. Table 1 lists the number of domains resulting in successful downloads during the first measurement iteration, numbers of successful downloads for following iterations only vary slightly within a deviation smaller than 3 %.

6.3 Comparison of Performance Indicators per Vantage Point

Conducting measurements from different vantage points indicates varying impacts on observed performance, for instance, due to different traversed Internet paths or distances between vantage points and measurement targets. Therefore, we survey performance indicators independent of the client configuration for the used vantage points. Figure 1 shows the cumulative distribution function (CDF) of mean throughput and mean RTT for the three vantage points for all considered option configurations.

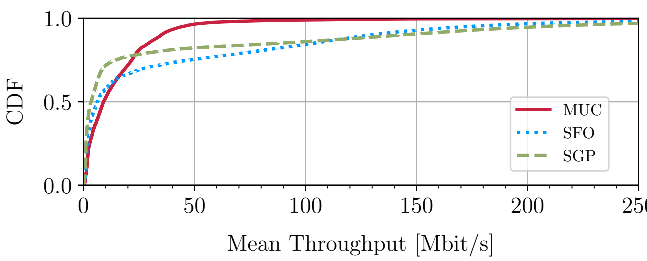
The VP in MUC results in a larger mean throughput up to about the 75th percentile of samples. Afterward, the VPs in SFO and SGP show larger shares of downloads with significantly increased mean throughput compared to the VP in MUC. This observation correlates to the distribution of mean RTTs. In particular, we find that the VPs in SFO and SGP result in a significant share of mean RTTs smaller than 5 ms, which is not observed for the VP in MUC.

However, the VP in MUC shows smaller mean RTTs for most of the remaining samples. We find that measurements conducted in SFO and SGP with mean RTTs smaller than 5 ms are associated

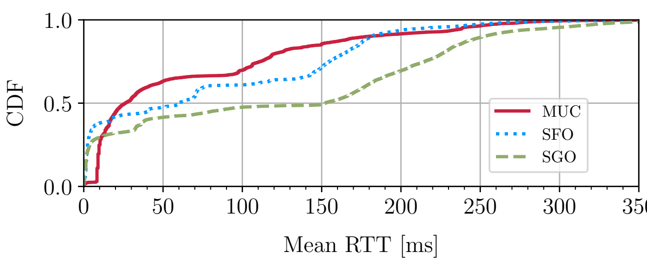
Table 2: Shares of downloads of a certain configuration (Config.) resulting in a certain decrease or increase of throughput compared to another configuration (vs.) aggregated for all VPs.

TCP options												
Config.	vs.	+	-	<0.5	0.5 - 0.6	0.6 - 0.7	0.7 - 0.8	0.8 - 0.9	0.9 - 1.0	1.0 - 1.1	1.1 - 1.2	>1.2
ECN	ALL14	9.6%	90.4%	39.3%	15.4%	8.5%	10.7%	9.7%	6.8%	3.2%	1.4%	5.1%
SACK	ALL14	10.3%	89.7%	37.7%	15.0%	8.8%	11.2%	10.2%	6.9%	3.4%	1.1%	5.8%
WS	ALL14	45.9%	54.1%	7.3%	1.7%	2.4%	3.9%	7.1%	31.7%	28.9%	5.3%	11.7%
ALL-WS7	ALL14	50.5%	49.5%	4.2%	1.1%	2.1%	3.1%	6.5%	32.5%	31.2%	6.3%	13.0%

QUIC and TCP												
Config.	vs.	+	-	0.7 - 0.8	0.8 - 0.9	0.9 - 1.0	1.0 - 1.1	1.1 - 1.2	1.2 - 1.3	1.3 - 1.5	1.5 - 2	>2
quiche	TCP-BL	72.1%	27.9%	2.8%	1.9%	3.1%	11.5%	7.1%	3.7%	6.4%	13.2%	30.2%
quiche	TCP-ALL14	47.6%	52.4%	3.9%	5.9%	5.4%	12.9%	6.5%	3.1%	5.6%	3.9%	15.7%



(a) Mean throughput.



(b) Mean RTT.

Figure 1: Measured KPIs per VP.

with measurement targets hosted by Cloudflare, Akamai, and partly Amazon.

6.4 Impact of TCP Options

To assess the impact of TCP option configuration on performance, we survey the CDF of mean throughput for each configuration, as shown in Figure 2.

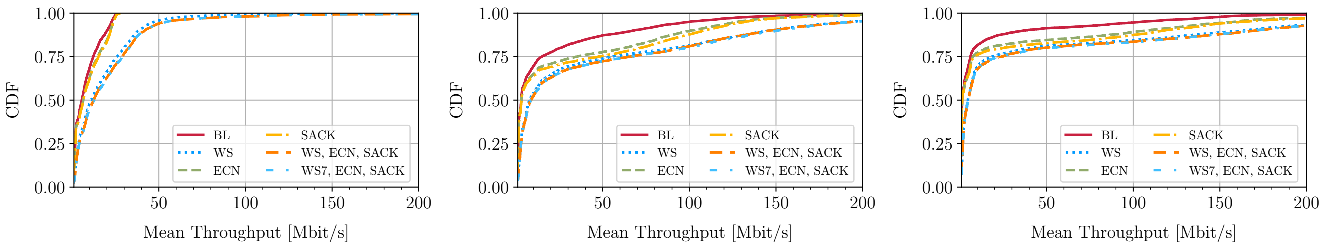
For the VP in MUC, we observe two groups of distributions: *(i)* configurations without enabled WS and *(ii)* configurations with enabled WS, while the latter indicates a significantly larger mean throughput. For measurements conducted with the VPs in SFO and SGP, we observe a significant increase of mean throughput measured for a subset of downloads with enabled ECN, respectively SACK, compared to the baseline configuration. Indicated performance gains can be traced back to downloads mainly conducted from targets hosted by Akamai, Amazon, and Cloudflare. However, analyzing the retransmission rates of the baseline measurements conducted on such targets reveals that the vast majority of downloads do not show any retransmission.

This observation implies that observed speed-ups are not caused by the impacts of ECN or SACK, which aim to avoid retransmissions, respectively, to mitigate the impact of lost packets. The baseline download is the first download conducted during each measurement run on a domain. Accordingly, observed speed-ups are likely caused by caching of downloaded files on edge servers of affected CDNs. Accordingly, the comparison between the baseline and other option configurations is biased. This bias does not apply to comparisons of other option configurations, as all downloads, despite the baseline, benefit from cached files. Note that we are currently conducting measurements considering a warm-up download before the actual baseline download to survey caching impacts in more detail. Results of such measurements will be published together with the implemented pipeline [39].

As the distribution of observed mean throughput does not show the explicit difference for two downloads of a run, we calculate the share of measurements with an option configuration resulting in a specific relative throughput change compared to the configuration supporting all options and a window scaling factor of 14 (TCP-ALL14). Table 2 shows shares of downloads resulting in a specific throughput change gathered across all measurement iterations and vantage points. As also observed for the CDFs of mean throughput, we find that measurements with TCP window scaling clearly outperform measurements without such an option. Nearly 40 % of measurements only conducted with ECN or SACK result in a throughput decrease larger than 50 % compared to TCP-ALL14.

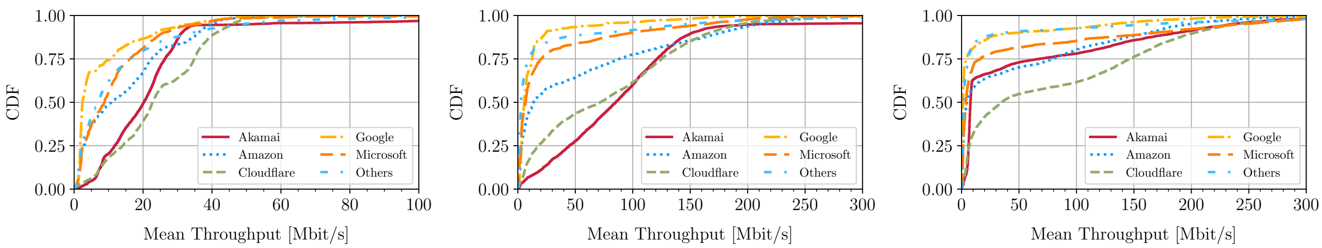
6.5 TCP vs. QUIC

Table 2 shows the speed-ups by downloads conducted with *quiche* compared to the mean throughput observed for TCP-BL and TCP-ALL14. Comparing the mean throughput of *quiche* to TCP-BL downloads results in positive speed-ups for over 70 % of measurements, while over 40 % of measurements show a speed-up by more than



(a) Vantage point in MUC. (b) Vantage point in SFO. (c) Vantage point in SGP.

Figure 2: Distribution of mean throughput measured with different TCP option configurations.



(a) Vantage point in MUC. (b) Vantage point in SFO. (c) Vantage point in SGP.

Figure 3: Distribution of mean throughput measured for domains grouped by their CDN affiliation.

50 %. In comparison to TCP downloads with all options enabled, *quiche* shows increased mean throughput for about 50 % of samples, while the shares of larger speed-ups are significantly smaller, as observed for the comparison of *quiche* and TCP-BL. Note that the first download of a run per domain during the QUIC measurements is conducted with *aioquic*, which implies bias due to caching at edge servers for the comparison of *aioquic* and *quiche*.

6.6 CDN Impact

To survey the impact of CDN hosting, we group measurement results according to the five considered giant CDNs and the sixth group, which includes all remaining domains. Figure 3 shows the CDF of mean throughput per domain for considered CDNs. The distribution of mean throughput shows that domains hosted by Cloudflare and Akamai provide the most significant shares of larger mean throughput, except for measurements to Akamai targets conducted by the VP in SGP. Domains hosted by Google and Microsoft show the least improved mean throughput compared to domains not hosted by one of the five giant CDNs.

Despite these observations, our measurements confirm the expectation that CDN hosting increases performance. The degree of performance gain for each CDN varies between the three vantage points, which is reasonable since the vantage point location determines the nearest point of presence (PoP) of a CDN.

Further measurements will be conducted to assess the impacts of edge server caching.

7 CONCLUSION

In this study, we conducted active Internet measurements with public web servers to assess the impact of TCP option usage, QUIC, and CDN hosting on connection performance. Our measurements show that TCP window scaling is crucial to increase throughput. Replacing TCP (using all options) with QUIC implies performance gain for nearly 50 % of samples. CDN hosting increases throughput for most considered CDNs despite Google compared to domains not hosted by one of the considered giant CDNs, while we observe varying performance depending on the vantage point.

For future work, we consider extending the introduced measurement pipeline to support additional protocol parameters, performance indicators, and analysis approaches like root cause analysis to determine the throughput limitations. Further measurement results to survey impacts by edge server caching will be published with the implemented pipeline [39].

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their valuable feedback. This work was partially funded by the German Federal Ministry of Education and Research under the project PRIMEnet (16KIS1370), 6G-life (16KISK002) and 6G-ANNA (16KISK107) as well as the German Research Foundation (HyperNIC, grant no. CA595/13-1). Additionally, we received funding by the Bavarian Ministry of Economic Affairs, Regional Development and Energy as part of the project 6G Future Lab Bavaria and the European Union's Horizon 2020 research and innovation program (grant agreement no. 101008468 and 101079774).

REFERENCES

[1] S. Bauer, B. Jaeger, F. Helfert, P. Barias, and G. Carle, "On the Evolution of Internet Flow Characteristics," in *Proceedings of the Applied Networking Research Workshop*, Jul. 2021.

[2] J. Mücke, M. Nawrocki, R. Hiesgen, P. Sattler, J. Zirngibl, G. Carle, T. C. Schmidt, and M. Wählisch, "Waiting for QUIC: On the Opportunities of Passive Measurements to Understand QUIC Deployments," *arXiv e-prints*, pp. arXiv–2209, 2022.

[3] A. Gerber and R. Doverspike, "Traffic types and growth in backbone networks," in *Optical Fiber Communication Conference*. Optica Publishing Group, 2011.

[4] I. Poese, B. Frank, B. Ager, G. Smaragdakis, and A. Feldmann, "Improving content delivery using provider-aided distance information," in *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, 2010, pp. 22–34.

[5] P. Gigis, M. Calder, L. Manassakis, G. Nomikos, V. Kotronis, X. Dimitropoulos, E. Katz-Bassett, and G. Smaragdakis, "Seven Years in the Life of Hypergiants' off-Nets," in *Proc. ACM SIGCOMM*, Virtual Event, USA, 2021.

[6] L. Zembruzki, R. Sommese, L. Z. Granville, A. Selle Jacobs, M. Jonker, and G. C. M. Moura, "Hosting Industry Centralization and Consolidation," in *2022 IEEE/IFIP Network Operations and Management Symposium (NOMS 2022)*. IEEE Press, 2022, p. 1–9.

[7] M. Kühlewind, S. Neuner, and B. Trammell, "On the State of ECN and TCP Options on the Internet," in *Passive and Active Measurement: 14th International Conference, PAM 2013, Hong Kong, China, March 18-19, 2013. Proceedings 14*. Springer, 2013, pp. 135–144.

[8] D. Murray, T. Koziniec, S. Zander, M. Dixon, and P. Koutsakis, "An analysis of changing enterprise network traffic characteristics," in *2017 23rd Asia-Pacific Conference on Communications (APCC)*. IEEE, 2017, pp. 1–6.

[9] H. Lim, S. Kim, J. Sippe, J. Kim, G. White, C.-H. Lee, E. Wustrow, K. Lee, D. Grunwald, and S. Ha, "A Fresh Look at ECN Traversal in the Wild," *arXiv preprint arXiv:2208.14523*, 2022.

[10] K. Nepomuceno, I. N. de Oliveira, R. R. Aschoff, D. Bezerra, M. S. Ito, W. Melo, D. Sadok, and G. Szabó, "QUIC and TCP: A performance evaluation," in *2018 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 2018, pp. 00 045–00 051.

[11] P. Biswal and O. Gnawali, "Does quic make the web faster?" in *2016 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2016, pp. 1–6.

[12] K. Wolsing, J. Rüth, K. Wehrle, and O. Hohlfeld, "A performance perspective on web optimized protocol stacks: TCP+ TLS+ HTTP/2 vs. QUIC," in *Proceedings of the Applied Networking Research Workshop*, 2019, pp. 1–7.

[13] D. A. Borman, R. T. Braden, and V. Jacobson, "TCP Extensions for High Performance," RFC 1323, May 1992. [Online]. Available: https://www.rfc-editor.org/info/rfc1323

[14] S. Floyd, J. Mahdavi, M. Mathis, and D. A. Romanow, "TCP Selective Acknowledgment Options," RFC 2018, Oct. 1996. [Online]. Available: https://www.rfc-editor.org/info/rfc2018

[15] S. Floyd, D. K. K. Ramakrishnan, and D. L. Black, "The Addition of Explicit Congestion Notification (ECN) to IP," RFC 3168, Sep. 2001. [Online]. Available: https://www.rfc-editor.org/info/rfc3168

[16] Y. Cheng, J. Chu, S. Radhakrishnan, and A. Jain, "TCP Fast Open," RFC 7413, Dec. 2014. [Online]. Available: https://www.rfc-editor.org/info/rfc7413

[17] J. Iyengar and M. Thomson, "QUIC: A UDP-Based Multiplexed and Secure Transport," RFC 9000, May 2021. [Online]. Available: https://www.rfc-editor.org/info/rfc9000

[18] M. Thomson and S. Turner, "Using TLS to Secure QUIC," RFC 9001, May 2021. [Online]. Available: https://www.rfc-editor.org/info/rfc9001

[19] J. Iyengar and I. Swett, "QUIC Loss Detection and Congestion Control," RFC 9002, May 2021. [Online]. Available: https://www.rfc-editor.org/info/rfc9002

[20] M. Allman, "A web server's view of the transport layer," *ACM SIGCOMM Computer Communication Review*, vol. 30, no. 5, pp. 10–20, 2000.

[21] K. Pentikousis and H. Badr, "Quantifying the deployment of TCP options - a comparative study," *IEEE Communications Letters*, vol. 8, no. 10, pp. 647–649, 2004.

[22] A. Medina, M. Allman, and S. Floyd, "Measuring interactions between transport protocols and middleboxes," in *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, 2004, pp. 336–341.

[23] C.-X. Chen and K. Nagaoka, "Analysis of the State of ECN on the Internet," *IEICE TRANSACTIONS on Information and Systems*, vol. 102, no. 5, pp. 910–919, 2019.

[24] M. Honda, Y. Nishida, C. Raiciu, A. Greenhalgh, M. Handley, and H. Tokuda, "Is it still possible to extend tcp?" in *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, 2011, pp. 181–194.

[25] K. Edeline and B. Donnet, "Evaluating the Impact of Path Brokenness on TCP Options," in *Proceedings of the Applied Networking Research Workshop*, 2020, pp. 38–44.

[26] Usage statistics of QUIC for websites. [Online]. Available: https://w3techs.com/technologies/details/ce-quic

[27] T. Shreedhar, R. Panda, S. Podanev, and V. Bajpai, "Evaluating QUIC Performance Over Web, Cloud Storage, and Video Workloads," *IEEE Transactions on Network and Service Management*, vol. 19, no. 2, pp. 1366–1381, 2021.

[28] Y. Yu, M. Xu, and Y. Yang, "When QUIC meets TCP: An experimental study," in *2017 IEEE 36th International Performance Computing and Communications Conference (IPCCC)*. IEEE, 2017, pp. 1–8.

[29] T. Mangla, E. Halepovic, M. Ammar, and E. Zegura, "emimic: Estimating http-based video qoe metrics from encrypted network traffic," in *2018 Network Traffic Measurement and Analysis Conference (TMA)*. IEEE, 2018, pp. 1–8.

[30] C. López, D. Morato, E. Magana, and M. Izal, "Effective analysis of secure web response time," in *2019 Network Traffic Measurement and Analysis Conference (TMA)*. IEEE, 2019, pp. 145–152.

[31] S. C. Madanapalli, H. H. Gharakhieli, and V. Sivaraman, "Inferring netflix user experience from broadband network measurement," in *2019 Network Traffic Measurement and Analysis Conference (TMA)*. IEEE, 2019, pp. 41–48.

[32] B. Jun, M. Varvello, Y. Zaki, and F. E. Bustamante, "WebTune: A Distributed Platform for Web Performance Measurements," in *2021 Network Traffic Measurement and Analysis Conference (TMA)*. IFIP, 2021.

[33] N. Zilberman, J. Woodruff, M. Ramanujam, A. W. Moore, Y. Tokusashi, D. A. Popescu, B. Cooper, P. Bressana, and S. Galea, "NRG: A Network Perspective on Applications' Performance," 2021.

[34] University of Oregon Route Views Project. [Online]. Available: https://www.routeviews.org/routeviews/

[35] A. Arturi, E. Carisimo, and F. E. Bustamante, "As2org+: Enriching AS-to-Organization Mappings With PeeringDB," in *Proc. Passive and Active Measurement (PAM)*, 2023, p. 400–428.

[36] D. Dittrich, E. Kenneally *et al.*, "The Menlo Report: Ethical principles guiding information and communication technology research," *US Department of Homeland Security*, 2012.

[37] C. Partridge and M. Allman, "Addressing Ethical Considerations in Network Measurement Papers," *Communications of the ACM*, vol. 59, no. 10, Oct. 2016.

[38] M. Koster, G. Illyes, H. Zeller, and L. Sassman, "Robots Exclusion Protocol," RFC 9309, Sep. 2022. [Online]. Available: https://www.rfc-editor.org/info/rfc9309

[39] S. Bauer, P. Sattler, J. Zirngibl, C. Schwarzenberg, and G. Carle. Tcp and quic measurement pipeline. [Online]. Available: https://github.com/tumi8/active-tcp-and-quic-measurements

[40] R. Fielding and J. Reschke, "Rfc 7230: Hypertext transfer protocol (http/1.1): Message syntax and routing," 2014.

[41] D. M. (aka Roadmaster), "Forcediphttpsadapter," Website, https://github.com/Roadmaster/forcediphttpsadapter; Last accessed: 12 Mai 2023.

[42] J. Lainé, "aioquic." [Online]. Available: https://github.com/aiortc/aioquic

[43] Cloudflare, "quiche." [Online]. Available: https://github.com/cloudflare/quiche

[44] B. Jaeger, J. Zirngibl, M. Kempf, K. Ploch, and G. Carle, "QUIC on the Highway: Evaluating Performance on High-Rate Links," in *International Federation for Information Processing (IFIP) Networking 2023 Conference (IFIP Networking 2023)*, Barcelona, Spain, Jun. 2023.

[45] Wireshark User's Guide - 7.5. TCP Analysis. [Online]. Available: https://www.wireshark.org/docs/wsug_html_chunked/ChAdvTCPAnalysis.html

[46] M. Prince, "Introducing: I'm under attack mode," Website, https://blog.cloudflare.com/introducing-im-under-attack-mode; Last accessed: 12 Mai 2023.

[47] About CrUX. [Online]. Available: https://developer.chrome.com/docs/crux/about/

[48] J. Zirngibl, P. Buschmann, P. Sattler, B. Jaeger, J. Aulbach, and G. Carle, "It's over 9000: Analyzing early QUIC Deployments with the Standardization on the Horizon," in *Proceedings of the 2021 Internet Measurement Conference*. New York, NY, USA: ACM, Nov. 2021.

[49] J. Zirngibl, P. Sattler, and G. Carle, "A First Look at SVCB and HTTPS DNS Resource Records in the Wild," in *International Workshop on Traffic Measurements for Cybersecurity 2023*, Jul. 2023.