Web Privacy By Design: Evaluating Cross-layer Interactions of QUIC, DNS and H/3

Jayasree Sengupta*, Mike Kosek†, Justus Fries†, Pratyush Dikshit*, and Vaibhav Bajpai*
*CISPA Helmholtz Center for Information Security, Germany
[jayasree.sengupta | pratyush.dikshit | bajpai]@cispa.de
†Technical University of Munich, Germany
[kosek@in.tum.de | justus.fries@tum.de]

***Abstract*—Every Web session involves a DNS resolution. While, in the last decade, we witnessed a promising trend towards an encrypted Web in general, DNS encryption has only recently gained traction with the standardisation of DNS over TLS (DoT) and DNS over HTTPS (DoH). Meanwhile, the rapid rise of QUIC deployment has now opened up an exciting opportunity to utilise the same protocol to not only encrypt Web communications, but also DNS. In this paper, we evaluate this benefit of using QUIC to coalesce name resolution via DNS over QUIC (DoQ), and Web content delivery via HTTP/3 (H3) with 0-RTT. We compare this scenario using several possible combinations where H3 is used in conjunction with DoH and DoQ, as well as the unencrypted DNS over UDP (DoUDP). We observe, that when using H3 1-RTT, page load times with DoH can get inflated by >30% over fixed-line and by >50% over mobile when compared to unencrypted DNS with DoUDP. However, this cost of encryption can be drastically reduced when encrypted connections are coalesced (DoQ + H3 0-RTT), thereby reducing the page load times by 1/3 over fixed-line and 1/2 over mobile, overall making connection coalescing with QUIC the best option for encrypted communication on the Internet.**

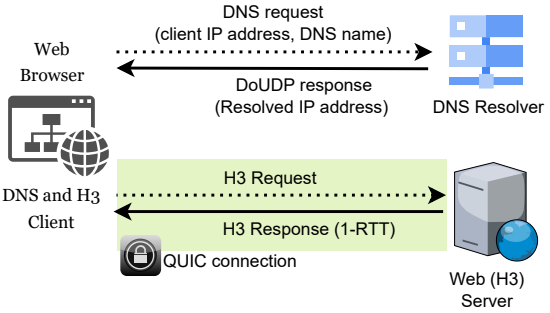
***Index Terms*—QUIC, Web, HTTP/3, DNS**

I. Introduction

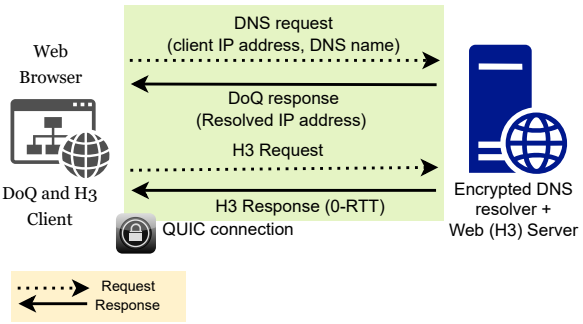
Over the last decade, with the increased privacy awareness amongst individuals, the Web slowly started becoming encrypted [1, 2]. However, encrypted DNS has only recently gained traction with the standardisation of DNS over TLS (DoT) [3] and DNS over HTTPS (DoH) [4], where in today's Internet unencrypted DNS resolution using DNS over UDP (DoUDP) remains the default [5]. Hence, despite the encryption of the actual Web content, the browsing behaviors of individuals can still be observed, enabling third parties to create trackable user profiles [6–9].

To counter this problem, today's browsers offer to encrypt DNS traffic using DoH [10], enabling users to opt-in into encrypted DNS with a public DNS resolver [11] of their choice. While DoH adds privacy to the DNS, hence enabling *Web Privacy By Design*, it remains rarely used, and is inherently limited by the underlying protocols: Multiple studies evaluate the impact of DoH and DoT on Web performance, finding that they are constrained by head-of-line blocking of the TCP connection, as well as the multiple round-trips required for the handshake of the TCP and TLS sessions [12–18].

To overcome these inherent limitations of TCP and TLS, the QUIC transport protocol has recently been standardized,



(a) Existing mechanism of Web Browsing with QUIC whereby a DNS request gets resolved over un-encrypted DoUDP followed by an encrypted HTTP/3 session.



(b) Proposed mechanism of Web Browsing whereby QUIC is used to coalesce name resolution with DoQ and Web content delivery with H3 0−RTT over a single QUIC connection.

Fig. 1: *Web Browsing over different unencrypted and encrypted DNS protocols using both H3 0-RTT and H3 1-RTT combinations.*

offering multiplexing support to address head-of-line blocking, and overcoming the handshake limitations by combining the transport and encryption handshake into a single round-trip [19]. Moreover, QUIC can also leverage 0−RTT in order to send application data within the first round-trip, effectively nullifying the handshake overhead altogether. QUIC was designed in tandem with HTTP/3 with focus on the encrypted Web: While H3 leverages QUIC as a transport protocol, requests can be multiplexed over a single QUIC connection, greatly reducing the overhead of HTTP/2 and HTTP/1.1 which

ISBN 978-3-903176-57-7

are required to establish multiple TCP and TLS sessions in order to avoid head-of-line blocking [20]. Hence, recent studies show that H3 improves over HTTP/2, finding reduced page load times (PLTs) for H3 while being less affected by packet loss and delay [21, 22], yet highlighting the importance of configuration choice for the performance of QUIC [23]. Moreover, encrypted DNS also benefits from QUIC, where the recently standardized DNS over QUIC (DoQ) [24, 25] improves over DoH and DoT [26]. Evaluating the impact on Web performance, it is shown that DoQ improves over DoH with up to 10% faster page loads on simple Web pages, and DoQ resulting in only 2% slower page loads in comparison to DoUDP on complex web pages.

Hence, QUIC greatly improves on the notion of *Web Privacy By Design*: where DoQ primarily benefits from faster handshakes, H3 avoids multiple handshakes by multiplexing requests over a single connection. Both protocols improve within their own layers, but the combination of DoQ and H3 significantly improves over DoH with HTTP/2. A typical Web browsing scenario over these protocols is depicted in Fig. 1a.

However, even when using QUIC for both DoQ and H3, the improvements are still uncoupled. Yet, CDN providers like Cloudflare offer both public DNS services using DoQ and Web content delivery using H3 on the same edge infrastructure [27]: Consequently, DNS resolution using DoQ, and preceding H3 requests to a web page hosted by the same CDN, will both be served using QUIC from the same infrastructure, offering optimization potential. The fresh H3 request to the web server happens over the same QUIC connection. This is exactly where our proposed QUIC connection coalescing is applicable as shown in Fig. 1b. For example, Cloudflare can majorly benefit from their existing setup to utilise QUIC to coalesce name resolution via DoQ and simultaneously execute Web content delivery using H3 with 0-RTT. By doing so, the Web communication is not only private but also becomes faster by reusing the same underlying QUIC connection.

In this paper, we evaluate the cross-layer interactions of QUIC, DNS, and H3, analyzing the benefits of using QUIC to coalesce name resolution with DoQ and Web content delivery with H3 $0-$RTT. To this end, we present a measurement setup that automates DNS resolution and Web browsing while emulating network conditions of a user at the edge based on real-world datasets for both fixed- and mobile-access network technologies. We find, that page load times using DoH can get inflated by $>30\%$ over fixed-line and by $>50\%$ over mobile when compared to unencrypted DNS with DoUDP, reflecting the cost of encrypted DNS using DoH. Taking *Web Privacy By Design* to the next level, we coalesce DoQ and H3 $0-$RTT connections, thereby reducing page load times by 1/3 over fixed-line and 1/2 over mobile in comparison to existing setup, overall making connection coalescing with QUIC the best option for encrypted communication on the Internet.

The remainder of this paper is structured as follows: We first present our methodology in Section II, followed by detailing our findings in Section III. Afterwards, Section IV discusses the Limitations and Future Work, after which we conclude the paper with Section V.

TABLE I: *Average values obtained from FCC's Measuring Broadband America and ERRANT datasets*

Access Technology	Delay (ms)	Download (Mbps)	Upload (Mbps)
Fiber	14.8	99.9	109.1
Cable	25.2	165.1	11.6
DSL	42.4	10.7	0.8
4G	91.9	54.0	21.2
4G medium	104.5	28.7	4.2

II. Methodology

To evaluate QUIC connection coalescing using DoQ + H3 $0-$RTT, our measurement setup automates DNS resolution and Web browsing while emulating network conditions of a user at the edge. It is based on real-world datasets for both fixed and mobile-access network technologies. Moreover, we compare this optimized approach to different combinations of H3 in conjunction with DoH and the unencrypted DoUDP due to their prevalence in today's browsers. To this end, the measurement setup decouples the DNS resolution from the actual web page loading on the client side, where the DNS and the H3 server run in the same process on the server side; as a design choice, we measure one DNS resolution to normalise the impact of DNS across different websites (see Section IV).

The measurement scenario is web browsing where *Chromium* [28] is used to measure page load times of three categories of web pages: an HTML page (example.org), an HTML page with javascript assets (wikipedia.org) and an HTML page with javascript assets, CSS and cookies (instagram.com). These web pages are chosen since they require only a single domain resolution to fully fetch the web page, i.e., all resources are fetched from the same host, and all HTTP requests are sent to it. To access a web page, first the domain name of the web page requested is resolved using DoQ, DoH, or DoUDP. Following, H3 is used to connect to the resolved IP address in order to directly fetch the content and render it within the browser. During this step, QUIC connection coalescing is simulated by using a QUIC $0-$RTT handshake within *Chromium's* H3 request, i.e., sending the HTTP request in conjunction with the first QUIC handshake packet.

The setup is encapsulated in Linux network namespaces, enabling separating client and server into different network domains. Following this, different network conditions are simulated using `netem` for fiber, cable, DSL and 4G. For 4G, two variations are used: 4G with good signal quality (referred to as 4G), as well as 4G with medium signal quality (4G medium). Table I shows the delay as well as bandwidth values that are applied for the different scenarios which are based on empirical data: FCC's Measuring Broadband America dataset [29] is used to represent the fixed broadband scenarios, whereas the ERRANT dataset [30, 31] is used for mobile wireless access technologies. The delays and bandwidth are controlled using `netem`, where delay is always meant in the sense of two-way delay, i.e., the round-trip time (RTT), where the on-way delay is assumed to be symmetrical.

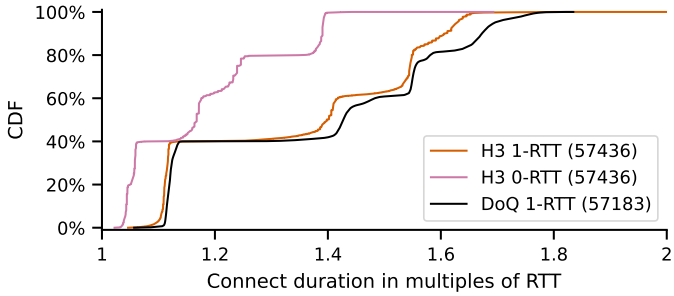


Fig. 2: *CDF of the QUIC handshake connect duration H3 for 1-RTT and 0-RTT, as well as DoQ 1-RTT for all scenarios. The values are normalized by the delay that was applied during the measurement to show how these metrics scale with round-trips.*

To enable this setup, several changes were made to the open source tools *CoreDNS* [32] and *Chromium*. *CoreDNS* was extended to additionally run an H3 server in order to share TLS information, resulting in an executable that runs both servers with the same certificates and *session ticket* keys. Moreover, *Chromium* was modified to support importing and exporting TLS session information, enabling 0-RTT and TLS session resumption following browser restarts.

In order to enable the reproduction of our findings, we have made the raw data of our measurements as well as the analysis scripts and supplementary files publicly available[1].

III. Evaluation

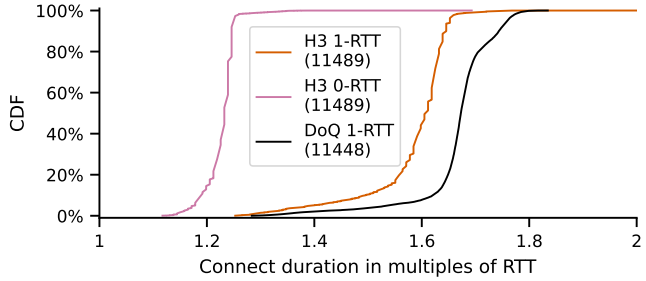
In order to evaluate QUIC connection coalescing, we first investigate the interaction of QUIC with DoQ and H3 in Section III-A, followed by an evaluation of the overhead of DoQ and DoH in comparison to the unencrypted DoUDP in Section III-B. Finally, we perform a detailed analysis of the web performance for the combination of all three DNS protocols with H3 1−RTT as well as 0−RTT, highlighting the benefits of QUIC connection coalescing in Section III-C.

A. On QUIC's Interaction with Application Layer Protocols

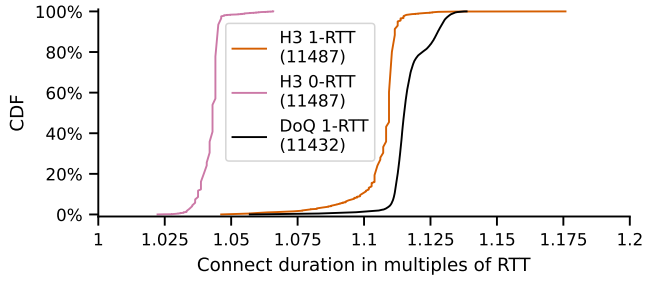
Within this section we illustrate how the QUIC handshake interacts with H3 as well as its scaling capability over various network conditions. As part of the evaluation, Fig. 2 shows two relevant metrics for H3: *connect* duration (i.e., *connectEnd* - *connectStart*)[2] and DoQ QUIC handshake duration measured in the DNS proxy. The *connect* duration is measured for both H3 with a 0-RTT and 1-RTT QUIC handshake. It is observed that H3 1-RTT *connect* times appear to roughly correspond to DoQ handshake times. This was verified by looking at *netlogs* and calculating the timespan between the client sending the initial and the last handshake packet (i.e., the `FIN` message), which appears to be at most around one millisecond lower than the reported *connect* time. This is the last message before the client sends its HTTP GET which means that the *connect* duration for 0-RTT accurately reflects the time it takes for the client to send its GET request. As a result, the H3 0-RTT

[1]https://github.com/justus237/DoQ-H3-analysis

[2]https://developer.mozilla.org/en-US/docs/Web/API/Performance_API/Navigation_timing



(a) Fiber scenario



(b) 4G Scenario

Fig. 3: *CDF of the QUIC handshake connect duration H3 for 1-RTT and 0-RTT, as well as DoQ 1-RTT. For fiber, the difference between HTTP 0-RTT and 1-RTT is large because the RTT is relatively low and thus the processing delay has a higher share. For 4G, the difference between 0-RTT and 1-RTT is small compared to other access technologies because the processing delay is small in proportion to the RTT.*

connect time is a valid metric to look at while measuring how long it takes until the first request is sent.

As expected, the plot shows that there is a difference between H3 0-RTT and 1-RTT of much less than one round-trip. The median for the *connect* duration of H3 0-RTT is 1.17 round-trips, which increases to 1.40 round-trips for 1-RTT (for comparison, DoQ has a median of 1.43 round-trips). However there is also a distinct step pattern visible in the distribution. While the values provided are normalized by the round-trip times for the access technologies, these steps are in fact caused by the difference between access technologies, meaning that the access technologies scale differently.

Figs. 3a and 3b reflect how the access technologies scale for fiber and 4G scenario respectively. It is observed from Fig. 3a that the distributions for *connect* times have a long tail in the high percentiles. 1-RTT shows a relatively large left tail from the minimum (i.e., 0th percentile, 1.25 round-trips) to around the 20th percentile (1.56 round-trips). We already know, the minimum for 0-RTT is 1.12 round-trips and the P20 value is 1.21 round-trips. As all data points are scaled by the same factor for a particular access technology, it means that the actual data itself for 0-RTT has less variation compared to 1-RTT. The median number of round-trips for 0-RTT is 1.23, which increases to 1.61 round-trips for 1-RTT (difference of 0.38 round-trips).

Comparing this observation to the difference in round-trips for 4G in Fig. 3b, we observe that the median for 1-RTT

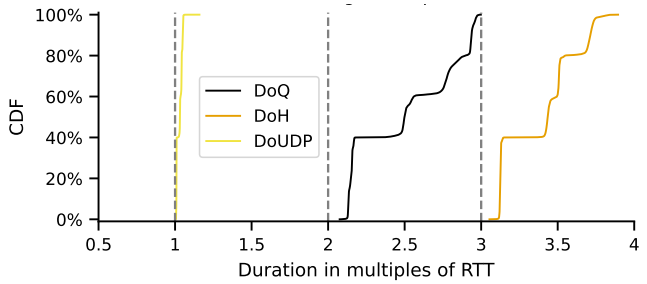


Fig. 4: *CDF of DNS exchange duration in multiples of round trip times for all scenarios. Only DoUDP scales with the number of expected round-trips.*

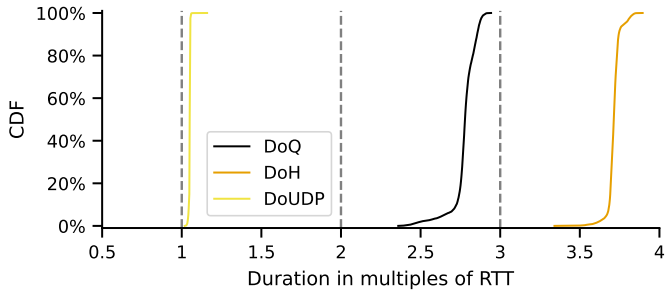
increases to 1.12 from 1.06 round-trips as for 0-RTT. The plot also shows how the different steps in Fig. 2 correspond to different access technologies despite normalizing by delay. Looking at the 0-RTT distribution, the step from P0 to P20 corresponds to the data shown in Fig. 3b. The step from P20 to P40 corresponds to 4G medium, the one from P40 to P60 is for cable, P60 to P80 is for fiber and lastly, P80 to P100 is for DSL. In addition to this, Fig. 3b also shows that 4G handshake time scales better with RTT while having less variation, thereby covering a smaller range of values. The minimum and maximum values for 0-RTT are 1.02 and 1.07 round-trips respectively.

Takeaway: *The overhead of client and/or server-side processing delay is relatively large for measurement setups where a low RTT access technology is emulated. While, in absolute terms, the processing delay is the same for access technologies with high RTTs, it weighs in much less relatively, resulting in the observed differences between H3 0-RTT and 1-RTT to be small in that case. However, 0-RTT still shows connect times.*

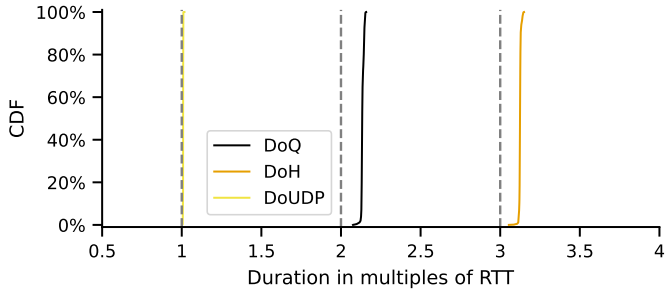
B. On DNS Overheads

To evaluate the overhead of DoQ and DoH in comparison to unencrypted DoUDP, we analyze the scaling factor for all the measured DNS protocols in terms of lookup times/exchange times (i.e. handshake times + query times). The data points are normalized by the scenario's delay where the expected values are: DoUDP does not require any connection setup round-trips, and we do not find any timeouts in our measurements; hence, the complete DNS exchange should take one round-trip in total. For DoQ, we assume QUIC `Address Validation Using Retry Packets` is disabled; hence, the DoQ handshake takes one round-trip. For DoQ, the handshake is without address validation which means it takes one round-trip. By adding the DNS query on top of that, DNS resolution then takes two round-trips in total. DoH is run with TLS 1.3 and thus the handshake takes two round-trips; adding the query time results in a total of three round-trips.

Fig. 4 shows the normalized lookups for all the three DNS protocols. It is observed from the plot that there are steps in the distribution for DoQ and DoH but not for DoUDP. The median for DoUDP is 1.03 round-trips whereas the maximum



(a) Fiber Scenario

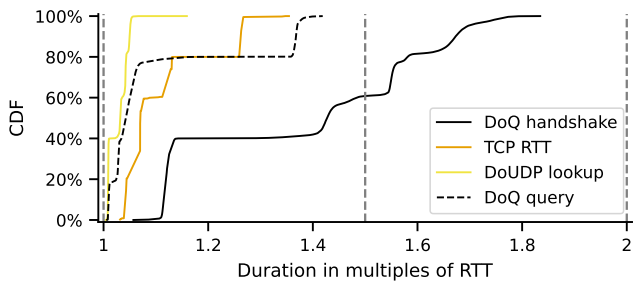


(b) 4G Scenario

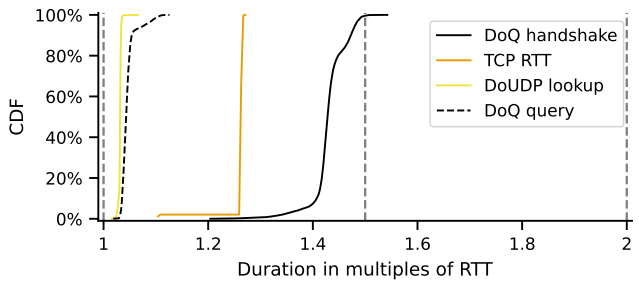
Fig. 5: *CDF of DNS exchange duration in multiples of RTT. Only DoUDP scales with the number of expected round-trips. The difference between DoQ and DoH is also one round-trip.*

is 1.16 round-trips. For DoQ, the median is 2.50 round-trips, the minimum is 2.07 round-trips and the maximum is 3.00 round-trips. For DoH, we see this increases by almost exactly one round-trip where the median is 3.43 round-trips having a minimum of 3.05 round-trips and a maximum of 3.89 round-trips. This means that while both DoQ and DoH do not appear to exhibit the expected number of round-trips for the whole DNS lookup, the difference between them is roughly one round-trip. The five steps in 20 percentile intervals are visible for DoQ as well as DoH and represent the different access technology scenarios. Since DoUDP scales with delay as per expectation, the overhead is likely not caused by any socket setup or network stack delay.

To confirm the above claim, Figs. 5a and 5b show the CDF of DNS exchange duration for the fiber and 4G setups respectively. The left tail for lower percentiles visible in the fiber plot for DoQ are also visible for DoH. The minimum (i.e., best case) for DoQ is 2.36 round-trips whereas for DoH it is 3.34 round-trips. The median, however, increases to 2.78 and 3.71 round-trips for DoQ and DoH respectively. Compared to 4G, the minimum for DoQ is 2.08 round-trips with a median of 2.13. For DoH, this increases by almost exactly one round-trip to 3.05 and 3.12 round-trips. This shows that the range of values for 4G is much smaller, meaning there is less variation in the data and there is no long tail as well. Analysing other access technology scenarios, the left tail appears to be the largest for fiber whereas it gets smaller when looking at scenarios with higher delay.



(a) All Scenarios



(b) Cable Scenario

Fig. 6: *CDF of TCP RTT, DoUDP lookups, DoQ queries, and DoQ handshakes, for all Scenarios. In theory, all these metrics (except for DoQ handshake durations) should take one round-trip.*

Finally, there exists one access technology where the difference between DoQ and DoH is not equivalent to one round-trip. Namely, in the case of DSL, the median of DoQ is 2.94 round-trips, while for DoH it is 3.51 round trips. This means that in this case, DoQ seems to have increased delay, despite the fact that Bandwidth-Delay Product (BDP) should be high enough. This increase is caused by higher than normal query duration. Note that the median DoQ query duration for DSL is 1.37 round-trips (min 1.35, max 1.42). For other access technologies the median is between 1 to 1.05 round-trips with no noticeable outliers for minimum or maximum values.

Digging deeper into this aspect, the measurement also contains data for the RTT of TCP (i.e., client sends a SYN and server responds with a SYN-ACK). The TCP round-trip times are analyzed to inspect whether the reason for the unusual scaling of DoH is rooted in something related to the TCP handshake or the TCP network stack itself. Since DoQ is run over UDP, the DoUDP can be used as the UDP socket setup time. The insights from above then indicate that at least for DoQ, the increased delay is not caused by anything related to the UDP stack and is likely caused by the QUIC stack.

Fig. 6a shows the TCP RTT, DoUDP lookup times, DoQ handshake times and DoQ query times. It is observed that for most of the data points, the scaling of DoUDP (median 1.03 RTTs), TCP RTT (1.07 RTTs) and DoQ query times (median 1.04 RTTs) are as expected. Explicitly, for DoQ query times, the increase for DSL is visible from P80 to P100.

There is also a noticeable increase in round-trips for this percentile range of TCP RTT. These data points belong to samples from the cable scenario, depicted in Fig. 6b. Here TCP RTT performs worse compared to both DoUDP lookups and DoQ query times across all percentiles. It is to be noted that the minimum value for TCP RTT is 1.10 round-trips, the median is 1.26 and the maximum is 1.27. On the contrary, DoUDP is at most 1.06 round-trips whereas DoQ queries are at most 1.13 round-trips.

> ***Takeaway:*** *DNS over QUIC shows expected improvements over DoH due its handshake requiring less RTTs, resulting in the DNS exchange duration of DoQ being roughly one round-trip faster in comparison to DoH for all scenarios except DSL. Moreover, lower RTT access technologies exhibit longer left tails, which eventually get smaller with increasing delay.*

C. On Interactions of H3 Across Different DNS Protocols

We perform experiments for three DNS protocols DoQ, DoH, and DoUDP, where DoH and DoUDP represent the encrypted and unencrypted DNS protocols commonly used in current web browsers. Each DNS protocol is combined with both H3 $0-$RTT and H3 $1-$RTT web performance measurements. A common web browsing scenario is defined as using DoUDP with H3 which is a realistic setup that likely provides the best performance with the caveat of DNS being unencrypted. DoQ with H3 $0-$RTT is referred to as QUIC *connection coalescing* as it represents the emulated optimized QUIC setup. Correspondingly, DoQ with H3 $1-$RTT is referred to as DoQ whereas DoH + H3 $1-$RTT is referred to as DoH. There are also permutations of DoUDP and DoH in combination with H3 $0-$RTT which are not investigated in this paper. The different access technology scenarios are not distributed evenly due to measurement interruptions. The sample sizes are as follows: fiber 68,934, DSL 68,928, 4G 68,922, cable 68,916 and 4G medium 68,916. For the same reason, the sample sizes for the measurements are also not distributed evenly: example.org 114,924, wikipedia.org 114,882 and instagram.com 114,810. These web pages were downloaded on June 8th, 2022 for the purpose of experimentation.

As DNS resolution is decoupled from the web browser, the DNS lookup time is added to the PLT web performance metric for H3 web performance measurement. Recall that one of our goals is to analyze how an optimized QUIC setup could perform. This is approximated by calculating the PLT for the setup where DoQ is used for DNS resolution and consequently *Chromium* is used to connect to the H3 server using a QUIC $0-$RTT handshake. Such a coalesced QUIC connection would take one round-trip for the initial QUIC connection (without address validation), another round-trip for the DNS query and a third round-trip for the H3 `SETTINGS` exchange. After that the actual H3 `GET` request and corresponding response takes place. Importantly, the `SETTINGS` exchange adds a round-trip because it is not implicitly done with the initial QUIC handshake or the DNS exchange. This results in three round-trips until the client sends its GET request, which is the same

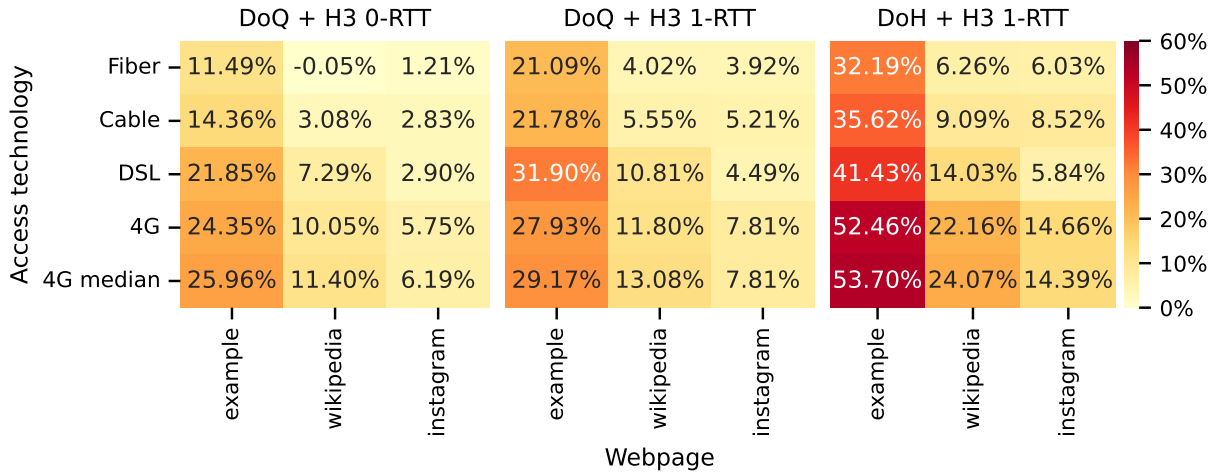


Fig. 7: *Heat map of relative PLT increase over DoUDP baseline for QUIC connection coalescing (i.e. DoQ + H3* $0-RTT$*), DoQ + H3* $1-RTT$*, and DoH + H3* $1-RTT$*. QUIC connection coalescing is equal to the baseline when replaying the* ***wiki page*** *over fiber.*

number of round-trips as the non QUIC coalescing scenario with DoQ and normal H3. This means that only the processing delay for the client and the server where they know the `SETTINGS` parameters beforehand and the server not having to send its certificate twice are subtracted from the overall web performance of normal H3 with DoQ.

The first set of experiment provides an overview of the median PLT increase for all the considered access technologies and web pages. The relative increase over the DoUDP + H3 $1-$RTT baseline is calculated for three protocol combinations: QUIC connection coalescing, DoQ and DoH. The relative increase is calculated using median values for both the protocol combinations (i.e., baseline and the comparator). The measurement is performed for a specific access technology and web page combination where the web pages are ordered by complexity horizontally. Note, the *example page* is a single HTML document whereas the *wikipedia page* includes Javascript in the HTML document to build the web page by fetching a single Javascript resource. On the contrary, the *instagram page* requires parsing and execution of seven Javascript resources (including React.js), two style sheets and finally produces a cookie popup banner while loading. The access technology scenarios are sorted by their delay vertically.

We observe from Fig. 7 that DoH setup has the highest relative increase across all web pages and access technologies. For the *example page* over 4G medium, it also has the overall worst case relative PLT increase of 53.7%. Additionally, for all the three protocol setups, the highest relative increase is observed for the *example page*. For almost all cases the relative increase for the *wikipedia page* is comparatively greater than that of the *instagram page*. This follows from the web page complexity as the *instagram page* is more resource-full and render time intensive than the *wikipedia page*. However, there is one exception to this for the fiber scenario of QUIC connection coalescing setup, whereby the PLTs for the simulated optimized QUIC connection coalescing setup are on average better than the DoUDP baseline by 0.05%. Lastly, we observe that for a lot of the web page columns, the performance of the access technologies degrade in an order of the respective RTT (delay). However, there are quite a few exceptions to this. For example, the relative increase for the DoQ setup over the baseline for the *example page* is highest in case of the DSL scenario as opposed to the 4G or the 4G medium one. On the other hand, loading the *instagram page* over DSL using the DoH setup (5.84%) observes lower relative increase than that of fiber (6.03%).

In the second set of experiment, we show the relative PLT increase in more detail. The distribution of the relative increase of all the PLTs (i.e., not just the median) over the median of DoUDP baseline are shown in Fig. 8. Note that in theory, the relative increase can be calculated using the value of the baseline for the same measurement run, since all protocol combinations are measured in every single run. However, the advantage of using the median is that the distribution of the data points relative to each other (data point represents frequency/probability) stays the same in comparison to the distribution of the absolute PLT values.

We observe that for the fiber scenario, measuring the *example page* over H3 $1-$RTT produces a distribution where there are two steps to the CDF along with two distinct PLT values that occur more frequently as opposed to a normal distribution centered around one value. This happens at the 60^{th} percentile, i.e., 60% of the data points are likely centered around one PLT value and the remaining 40% around another, higher one. To dig deeper, we investigate the other web performance metrics. It is observed from the data that this split in values is first visible for the *domInteractive* metric. Before that, *responseEnd* doesn't have split values. This means that the root cause behind such distinct central values is not related to fetching the web page, instead they are a result of building the Document Object Model (DOM). Additionally, this happens when *gzip* is disabled and not from decoding the HTML document.

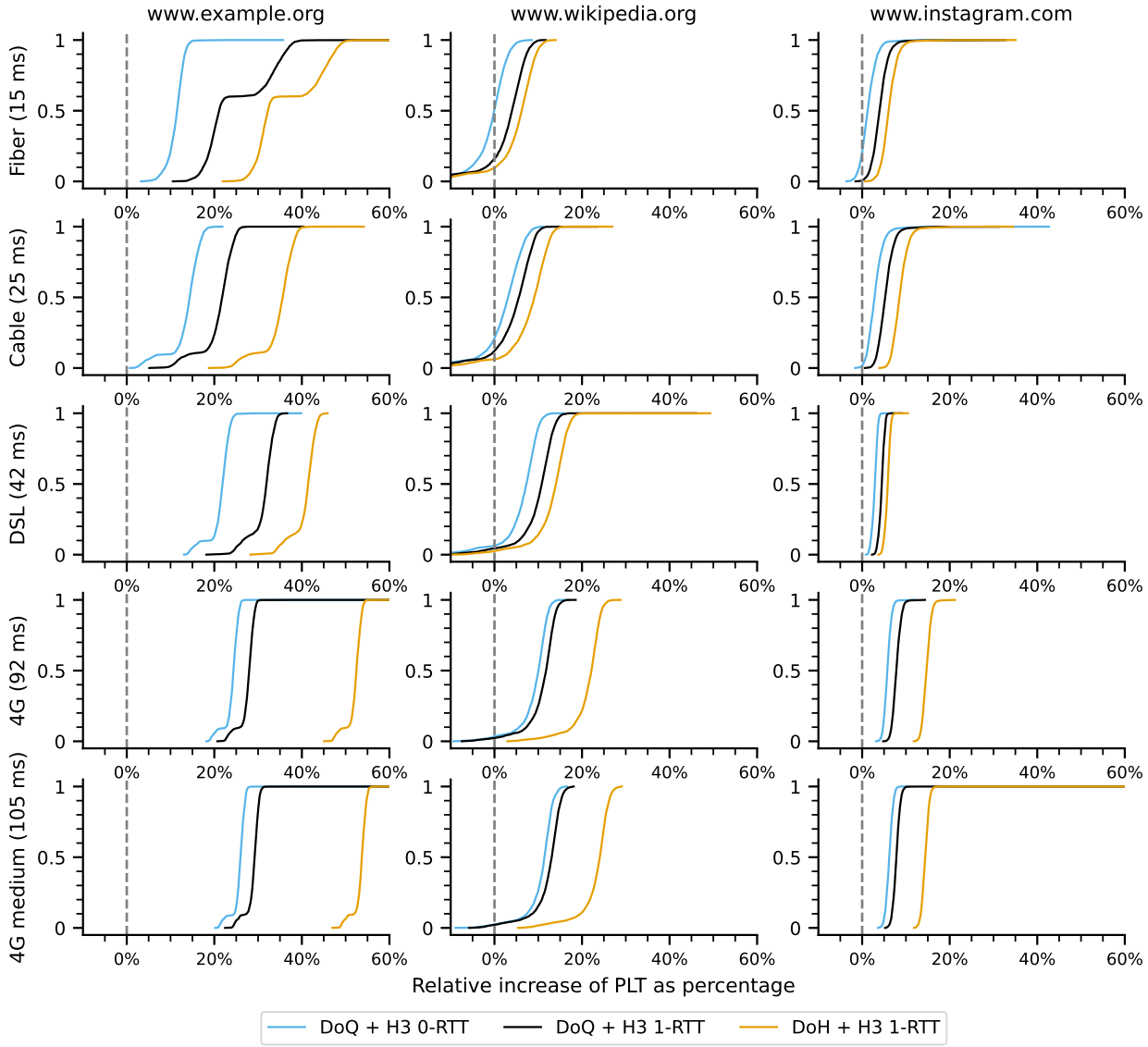


Fig. 8: *Grid of CDFs showing the relative increase of QUIC connection coalescing (i.e. DoQ + H3* $0-$*RTT), DoQ + H3* $1-$*RTT, and DoH + H3* $1-$*RTT over the horizontal DoUDP baseline for the five access technologies and the three web pages. Relative changes between the protocol combinations are affected by both of these dimensions.*

Another observation specific to the *example page* is that for all access technologies excluding fiber, there is a short left tail in the distribution upto the 10^{th} percentile. For example, in case of cable the P10 relative increase for DoQ scenario is 14.5%, while the P20 value is 19.6% and the corresponding median is 21.8%. These tails are a result of both the handshake time having left tails, as shown above along with the time it takes to fetch additional resources plus the rendering time. For example, the distributions of the time between *responseEnd* and *loadEventStart* has similar short left tails. For the *wikipedia page* there is a longer left tail compared to the *example page* across all access technologies, however for the *instagram page*, there is no left tail visible at all.

Overall, Fig. 8 demonstrates that both dimensions (i.e. web page and access technology) have an effect on the relative increase over the DoUDP baseline as well as the difference between the protocol setups. Specifically for the simplest web page, i.e. the *example page*, the differences in percentage points between the protocol combinations are the largest, and for the *instagram page*, the differences between them are significantly reduced. This apparently happens as the time spent by the browser in parsing the HTML documents, building the DOM and executing Javascript increases, henceforth the DNS and H3 connection setup times have less influence on the total PLT. With increasing complexity of the web page, the potential time saving (in relation to the time it takes to load a page) from changing the underlying protocols used for DNS and H3 significantly decreases.

The difference between DoQ and DoH scales with the round-trip time (except for the DSL measurement, see § III-B).

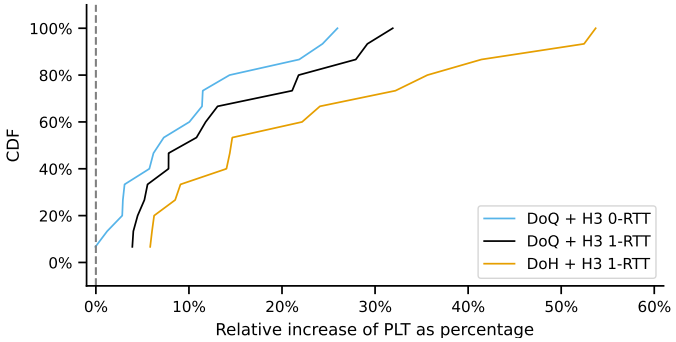


Fig. 9: *CDF of the relative increase of protocol combinations over the DoUDP baseline.*

However, the difference between H3 0−RTT and 1−RTT does not, as can be seen in Fig. 8 as well. For instance, observing the fiber scenario with the lowest round-trip time for the *wikipedia page*, the difference in medians between the QUIC connection coalescing setup and DoQ is 4.0 percent points. On the other hand, the difference between the medians of DoQ and DoH is 2.3 percent points. However, with increasing round-trip times (i.e., CDFs below fiber in the same column), the percentage point difference between DoQ and DoH increases. For example, in case of 4G, it increases to 10.4 percent points, while the difference in medians between DoQ and QUIC connection coalescing decreases to 1.8 percent points. The same effect is visible in the distributions for the *instagram page* where fiber 0−RTT (at the median) scenario saves 2.7 percent points while transitioning from DoH to DoQ saves 2.1 percent points. For 4G, these values are 1.6 percent points and 6.6 percent points respectively. Since all data points within a CDF are scaled by the same median value, this observation also holds for the absolute PLTs.

Overall, these observations mean that with increasing delay between the client and server, the potential time savings (relative to the PLT) of 0−RTT decreases, while the savings for using DoQ instead of DoH increases as time spent by the browser in rendering is less affected by delay. However, it is still slightly affected by delay because of resources that need to be fetched after the base HTML document is retrieved.

To summarize, Fig. 9 shows as a CDF, the relative PLT increase (at the median) for the relevant protocol combinations to the DoUDP baseline. Each protocol combination has 15 data points in the CDF, one for each *[web page, access technology]* tuple. As explained above, the baseline is a common web browsing scenario over unencrypted DNS. The QUIC connection coalescing setup can only match it for one tuple where the median relative increase is 7.3%. For a DoQ setup, the median is slightly higher at 10.8%. Finally the DoH setup, which is a protocol combination that is present in *Chromium* right now, has an average relative increase of 14.7%. In the worst case, QUIC connection coalescing exhibits an increase of 26.0%, DoQ at 31.9% and DoH at 53.7% respectively.

The percentage point difference between DoH and DoQ in the worst case is much larger than the one between DoQ and QUIC connection coalescing. This means that for worst case scenarios, an end-user can drastically improve their performance by using DoQ. On the contrary, the end-user gains relatively less performance under a unified QUIC connection for DNS and H3. This, however, comes with the caveat that 0−RTT does not actually save a full round-trip due to H3's `SETTINGS` exchange. If this exchange were made earlier, e.g., by piggybacking the DNS request and response or even the initial QUIC handshake, a full round-trip could be saved, thereby making the performance closer to the baseline DoUDP + H3 1-RTT setup. However, out of the encrypted DNS protocols, QUIC connection coalescing setup is still the best option for a fast private web browsing experience.

> ***Takeaway:*** *Using H3 1−RTT, page load times with DoH can get inflated by >30% over fixed-line and by >50% over mobile compared to unencrypted DoUDP. However, cost of encryption is substantially reduced when encrypted connections are coalesced using DoQ + H3 0−RTT, thereby reducing the page load times by 1/3 over fixed-line and 1/2 over mobile compared to the existing setup. Overall, our findings show that QUIC connection coalescing is the best option for encrypted communication on the Internet.*

IV. Limitations and Future Work

There are a few noticeable limitations. First, the presented findings represent an emulated setup where the DNS name resolution had to be decoupled from the web browsing process. Secondly, the use case of measuring an HTML page over an emulated fiber connection shows that the page load times have two central values. While considering all web performance metrics, we find that this split happens after the web page is already fetched while building the DOM. Yet, we were not able to investigate the root cause of this behavior. The measurement setup to evaluate QUIC connection coalescing using DOQ + H3 for 0−RTT is limited to web pages having a single DNS resolution. As such, the setup itself is currently implemented with a single H3 web server that serves as a directory to replay web pages. However, all resources being served by the same host is an uncommon scenario on the Web, since most web pages use third-party resources. Moreover, for websites with several DNS resolutions, a scaling factor can be applied to the results presented in the paper.

We plan to further refine the introduced concept of QUIC connection coalescing in the future. For instance, *Chromium* will be extended with support for DoQ in order to couple DNS resolution with web browsing, resulting in a measurement setup capable of QUIC connection coalescing. This will also extend the methodology to web pages with more than one DNS resolution, enabling the measurement of arbitrary web pages. We also plan to extend the setup to emulate packet loss and cross-traffic network conditions. Finally, while we use DoH with HTTP/2 as the current de-facto standard for encrypted DNS on the web, DNS over HTTP/3 (DoH3) is expected to gain traction in the coming month. Though not

widely supported, Google has added DoH3 to their public DNS service as well as Android in July 2022 [33]. Cloudflare has also added DoH3 support to their public DNS service in March 2022 [34]. Hence, we plan to extend our work with DoH3 further by blurring the boundaries between DNS resolution and Web content delivery.

V. Conclusion

In this paper, we evaluated the cross-layer interactions of QUIC, DNS, and H3, highlighting the benefits of using QUIC to coalesce name resolution via DNS over QUIC and Web content delivery via H3 with $0-$RTT. With the introduced measurement setup, we performed automated measurements of DNS resolution and Web browsing while emulating network conditions based on real-world datasets for both fixed-line and mobile-access network technologies. Our findings show that page load times using DNS over HTTPS can get inflated by $>30\%$ over fixed-line and by $>50\%$ over mobile when compared to unencrypted DNS over UDP, reflecting the cost of encrypted DNS. Taking *Web Privacy By Design* to the next level, we coalesced DNS over QUIC and H3 $0-$RTT connections. With reduced page load times by 1/3 over fixed-line and 1/2 over mobile compared to existing Web browsing setup, our findings highlight that QUIC connection coalescing is currently the best option for encrypted communication on the Internet.

Acknowledgment

This work was supported by the Volkswagenstiftung Niedersächsisches Vorab (Funding No. ZN3695).

References

[1] C. Chan *et al.*, "Monitoring TLS adoption using backbone and edge traffic," in *IEEE INFOCOM 2018*, 2018. [Online]. Available: https://doi.org/10.1109/INFCOMW.2018.8406957

[2] M. Trevisan *et al.*, "Five Years at the Edge: Watching Internet from the ISP Network," in *CoNEXT*. ACM, 2018, pp. 1–12. [Online]. Available: https://doi.org/10.1145/3281411.3281433

[3] Z. Hu *et al.*, "Specification for DNS over Transport Layer Security (TLS)," *RFC*, vol. 7858, pp. 1–19, 2016. [Online]. Available: https://doi.org/10.17487/RFC7858

[4] P. E. Hoffman and P. McManus, "DNS queries over HTTPS (doh)," *RFC*, vol. 8484, pp. 1–21, 2018. [Online]. Available: https://doi.org/10.17487/RFC8484

[5] C. Deccio and J. Davis, "DNS Privacy in Practice and Preparation," in *CoNEXT 2019*, 2019. [Online]. Available: https://doi.org/10.1145/3359989.3365435

[6] D. W. Kim and J. Zhang, "You Are How You Query: Deriving Behavioral Fingerprints from DNS Traffic," in *SecureComm 2015*, 2015. [Online]. Available: https://doi.org/10.1007/978-3-319-28865-9_19

[7] M. Kirchler *et al.*, "Tracked Without a Trace: Linking Sessions of Users by Unsupervised Learning of Patterns in Their DNS Traffic," in *AISec 2016*, 2016. [Online]. Available: https://doi.org/10.1145/2996758.2996770

[8] J. Li *et al.*, "Can We Learn what People are Doing from Raw DNS Queries?" in *INFOCOM 2018*, 2018. [Online]. Available: https://doi.org/10.1109/INFOCOM.2018.8486210

[9] L. Zhu *et al.*, "Connection-Oriented DNS to Improve Privacy and Security," in *IEEE Symposium on Security and Privacy 2015*, 2015. [Online]. Available: https://doi.org/10.1109/SP.2015.18

[10] DNS Privacy Project, "Public Resolvers," 2022, [Online; accessed 2022-Oct-12]. [Online]. Available: https://dnsprivacy.org/public_resolvers/

[11] T. V. Doan *et al.*, "Evaluating Public DNS Services in the Wake of Increasing Centralization of DNS," in *IFIP Networking Conference*. IEEE, 2021, pp. 1–9. [Online]. Available: https://doi.org/10.23919/IFIPNetworking52078.2021.9472831

[12] Trinh Viet Doan *et al.*, "Measuring DNS over TLS from the Edge: Adoption, Reliability, and Response Times," in *PAM 2021*, 2021. [Online]. Available: https://doi.org/10.1007/978-3-030-72582-2_12

[13] A. Hounsel *et al.*, "Can Encrypted DNS Be Fast?" in *PAM 2021*, 2021. [Online]. Available: https://doi.org/10.1007/978-3-030-72582-2_26

[14] C. Lu *et al.*, "An End-to-End, Large-Scale Measurement of DNS-over-Encryption: How Far Have We Come?" in *IMC 2019*, 2019. [Online]. Available: https://doi.org/10.1145/3355369.3355580

[15] R. Chhabra *et al.*, "Measuring DNS-over-HTTPS Performance around the World," in *IMC 2021*, 2021. [Online]. Available: https://doi.org/10.1145/3487552.3487849

[16] A. Hounsel *et al.*, "Comparing the Effects of DNS, DoT, and DoH on Web Performance," in *WWW 2020*, 2020. [Online]. Available: https://doi.org/10.1145/3366423.3380139

[17] T. Böttger *et al.*, "An Empirical Study of the Cost of DNS-over-HTTPS," in *IMC 2019*, 2019. [Online]. Available: https://doi.org/10.1145/3355369.3355575

[18] K. Borgolte *et al.*, "How DNS over HTTPS is Reshaping Privacy, Performance, and Policy in the Internet Ecosystem," in *TPRC47 2019*, 2019. [Online]. Available: https://doi.org/10.2139/ssrn.3427563

[19] J. Iyengar and M. Thomson, "QUIC: A UDP-Based Multiplexed and Secure Transport," *RFC*, vol. 9000, pp. 1–151, 2021. [Online]. Available: https://doi.org/10.17487/RFC9000

[20] M. Kosek, T. Shreedhar, and V. Bajpai, "Beyond QUIC v1: A First Look at Recent Transport Layer IETF Standardization Efforts," *IEEE Communications Magazine*, vol. 59, no. 4, pp. 24–29, 2021. [Online]. Available: https://doi.org/10.1109/MCOM.001.2000877

[21] M. Kosek *et al.*, "Exploring Proxying QUIC and HTTP/3 for Satellite Communication," in *IFIP Networking 2022*, 2022. [Online]. Available: https://doi.org/10.23919/IFIPNetworking55013.2022.9829773

[22] T. Shreedhar *et al.*, "Evaluating QUIC Performance Over Web, Cloud Storage, and Video Workloads," *IEEE Transactions on Network and Service Management*, vol. 19, no. 2, pp. 1366–1381, 2022. [Online]. Available: https://doi.org/10.1109/TNSM.2021.3134562

[23] A. Yu and T. A. Benson, "Dissecting Performance of Production QUIC," *WWW Conference*, 2021. [Online]. Available: https://doi.org/10.1145/3442381.3450103

[24] C. Huitema *et al.*, "DNS over Dedicated QUIC Connections," RFC 9250, 2022. [Online]. Available: https://www.rfc-editor.org/info/rfc9250

[25] M. Kosek *et al.*, "One to Rule Them All? A First Look at DNS over QUIC," in *Passive and Active Measurement Conference*, 2022. [Online]. Available: https://doi.org/10.1007/978-3-030-98785-5_24

[26] M. Kosek, L. Schumann, R. Marx, T. V. Doan, and V. Bajpai, "DNS Privacy with Speed? Evaluating DNS over QUIC and Its Impact on Web Performance," in *Proceedings of the 22nd ACM Internet Measurement Conference*, 2022, p. 44–50. [Online]. Available: https://doi.org/10.1145/3517745.3561445

[27] Cloudflare, "Unimog - Cloudflare's edge load balancer," 2022, [Online; accessed 2022-Oct-12]. [Online]. Available: https://blog.cloudflare.com/unimog-cloudflares-edge-load-balancer/

[28] T. C. Projects, "Chromium open-source browser," 2022. [Online]. Available: https://www.chromium.org/Home

[29] J. Fries *et al.*, "An Eight Years Perspective on the Internet Broadband Infrastructure in the USA," in *IFIP Networking*, 2022. [Online]. Available: https://doi.org/10.23919/IFIPNetworking55013.2022.9829788

[30] M. Trevisan *et al.*, "ERRANT: Realistic emulation of radio access networks," *Computer Networks*, 2020. [Online]. Available: https://doi.org/10.1016/j.comnet.2020.107289

[31] C. Midoglu *et al.*, "MONROE-Nettest: A configurable tool for dissecting speed measurements in mobile broadband networks," in *INFOCOM 2018*, 2018. [Online]. Available: https://doi.org/10.1109/INFCOMW.2018.8406836

[32] AdGuard, "CoreDNS fork for AdGuard DNS," 2022, [Online; accessed 2022-Oct-12]. [Online]. Available: https://github.com/AdguardTeam/coredns

[33] Google, "DNS-over-HTTP/3 in Android," 2022, [Online; accessed 2022-Oct-12]. [Online]. Available: https://security.googleblog.com/2022/07/dns-over-http3-in-android.html

[34] Cloudflare, "Cloudflare Blog: Announcing experimental DDR in 1.1.1.1," 2022, [Online; accessed 2022-Oct-12]. [Online]. Available: https://blog.cloudflare.com/announcing-ddr-support/