This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2024.3383787

On Cross-Layer Interactions of QUIC, Encrypted DNS and HTTP/3: Design, Evaluation and Dataset

Jayasree Sengupta*, Mike Kosek†, Justus Fries†, Simone Ferlin-Reiter‡, and Vaibhav Bajpai§
*CISPA Helmholtz Center for Information Security, Germany [jayasree.sengupta@cispa.de]
†Technical University of Munich, Germany [kosek@in.tum.de | justus.fries@tum.de]
‡Red Hat and Karlstad University, Sweden [simone@ferlin.io]
§Hasso Plattner Institute, Germany [vaibhav.bajpai@hpi.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.**

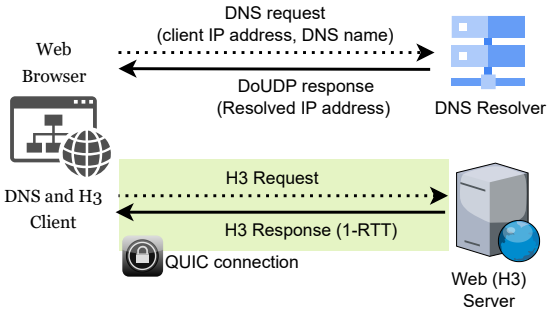
***Index Terms*—QUIC, Web, HTTP/3, DNS**

I. Introduction

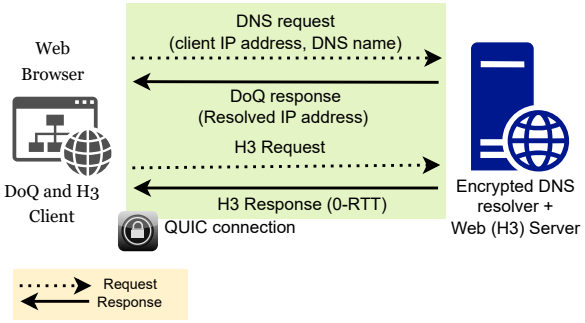
Over the last decade, with the increased privacy awareness amongst individuals, the Web slowly started becoming encrypted [1, 2]. However, encrypted DNS has only recently gained traction with the standardisation of DNS over TLS (DoT) [3] and DNS over HTTPS (DoH) [4], where in today's Internet unencrypted 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 are required to establish multiple TCP and TLS sessions

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2024.3383787

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] improves over DoH and DoT [25][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. Overall, we provide two main contributions:

- **Measurement Method –** 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. We hereby present a measurement setup (see: § III) 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.
- **Findings –** We show (see: § IV) 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 [28]. 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. 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 available [29].

This paper builds on our earlier work [30]. In this paper, we have added substantial background material, including a review (see: § II) of recent web performance testing, monitoring and related methods over all three encrypted DNS protocols (DoT, DoH and DoQ), QUIC and HTTP/3. We have added further details to our methodology with illustrations of the measurement setup (see: § III) to aid the readership. In addition, we have added new results, encompassing a detailed analysis (see: § IV-D) on two categories of websites: (a) HTML Page with Javascript (b) HTML Page with Javascript and CSS. We have also added a new section further highlighting the broader implications (see: § V) of our work, and discussing new research directions. Our results establish that QUIC connection coalescing is the best option for encrypted communication on the Internet, however, performance gains vary depending on the website and access technology combination used. Towards the end, we discuss § VI limitations and future scope, followed by the concluding remarks in § VII.

II. Background and Related Work

In the following we introduce the three main protocols studied in this paper: DNS with its three most recent secured variations - DoT, DoH and DoQ - HTTP3, and QUIC.

A. DNS protocol

In essence, the DNS protocol is the Internet's phonebook, where a user asks a DNS resolver to translate a human-readable domain name, e.g., business.com into a machine-readable IP address and vice-versa. The DNS protocol typically uses port 53, also known as Do53, and supports unencrypted queries over both UDP and TCP protocols. Under the hood, the DNS protocol is a distributed system composed of a global network of nameservers organised as an hierarchical database of resource records. Typically, a user sends out a request for resource record to a recursive resolver, typically operated by network providers, acting as a proxy. If the resource record is already in this resolver's cache, the reply is sent straight back to the user. If it is not, the resolver will, starting from the root, traverse the hierarchical tree of nameservers until it receives an authoritative answer to the user's resource record request.

In 1983, when the DNS protocol was introduced, privacy did not have the same consideration it has today [31, 32]. For this reason, DNS messages are typically sent in plain text to the recursive resolvers. Unencrypted DNS messages reveal a great deal of the user's behaviour in the Internet, allowing anyone on the path between the user and nameserver to eavesdrop or make use of DNS for distributed Denial-of-Service (DDoS) attacks. In today's Internet, the DNS protocol also became a critical piece, due to its relevance for Content Distribution Networks (CDNs) for traffic redirection. With the development of Internet censorship and surveillance mechanisms, privacy considerations has been inevitably included in modern protocol standards' development, which includes more recent attempts to secure the DNS protocol, namely, DNS over TLS (DoT), DNS over HTTPS (DoH), and DNS-over-QUIC (DoQ).

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2024.3383787

DoT [3] establishes a TLS session between the user and the recursive resolver on port 853, to exchange subsequent encrypted DNS queries and responses. This portion of the DNS request, i.e., the path between user and the recursive resolver, can easily be associated with individual users. For this reason, but not limited to this portion of the DNS request, DoT can be seen as an attempt to primarily provide privacy for the portion of the DNS request between the user and the recursive resolver. DoH [4] runs atop TCP on port 443, which is the standard port for HTTPS, i.e., Internet traffic. By using HTTPS, DoH traffic inherently looks like any other encrypted Internet traffic. Thus, DoH has been considered more robust against censorship mechanisms or port-based firewalls compared to DoT. Then, DoH sends DNS requests in an HTTP GET request on HTTPS default port 443. DoQ [24] is a third attempt to improve DNS request privacy and minimize latency by leveraging QUIC as the underlying protocol. Although encrypted DNS protocols such as DoT and DoH are already deployed and in use, they suffer from shortcomings due to being based on TCP. In other words, although DoQ carries privacy properties similar to DoT and DoH, the latency characteristics of DoQ is more similar to the unencrypted DoUDP.

With several encrypted DNS standards available, research has been looking at unencrypted DNS [33, 34] and also comparing it with DoT, DoH and, more recently, DoQ (see: Table I). In 2021, [35] shows that although the amount of Internet traffic for encrypted DNS was not growing, there was a growing number of DoH servers available - for benign and malicious purposes. [36] provides an overview of DNS encryption proposals, discussing the value of the protection dependent on the trust of end users in the DNS resolvers. In [37], the authors perform a trace file analysis with DNS traffic over two research institute networks looking at the performance of DNS requests, failures, errors, caching effectiveness. [13] studies the performance of encrypted DNS versus unencrypted DNS in home networks, where DoT obtained lower latency compared to DNS whereas DoH had significant performance variation depending on the recursive resolver. Then, [15] confirmed that performance of DoH varies, looking at geographic differences compared to unencrypted DNS. In [26], the authors looks at DoQ, showing a steady increase in adoption, with a good portion of the measurements indicating higher handshake times, however, with DoQ still outperforming DoT and DoH. Further, [38] surveys the DNS encryption standards and literature between 2016 to 2021 looking at their adoption status, performance, benefits, and security issues. Then, the authors show the current landscape, how encrypted DNS is misused by malware, and also highlight DNS traffic inference techniques currently available. In [39] the authors also look at how much traffic analysis can deduct from DoT messages, confirming that information leakage is possible even when DoT messages are padded.

B. QUIC protocol

The most recent transport layer revolution has been undoubtedly the QUIC protocol [19]. With the promise of being more simply extendable, maintainable and deployable, QUIC is a connection-oriented, end-to-end encrypted transport protocol based on UDP.

With growing interest for QUIC in general, there has been research (see: Table I) evaluating the protocol considering different network scenarios and technologies such as wireless, satellite networks and IoT, server and client stacks, configuration and location. Already in 2017, [40] compares QUIC with respect to the network, the website structure and involved end-to-end actors. Then, [41] confirmed that QUIC traffic already in 2018 accounted to up to 9% of the Internet traffic.

In [42], the authors study the performance of the Message Queuing Telemetry Transport Protocol (MQTT) over QUIC, where the authors confirm good performances for typical IoT use cases. [43] proposes two different cross-layer approaches to compares against QUIC over wireless networks, while [44] looks at QUIC performance in wireless mesh networks. In [23], the authors compare the performance of QUIC and TCP against production servers hosted by Google, Facebook, and Cloudflare under several network conditions, applications, and client implementations, reporting performance benefits of QUIC largely linked to the QUIC server and client configurations such as congestion control and stack tuning.

Then, [22] evaluates QUIC performance over Internet transfers, cloud storage, and video applications, and it compares it against TLS/TCP. The authors confirm lower latencies for Internet transfers over QUIC, in cloud storage with certain file sizes, and with video streaming. In [45], the authors look at QUIC connection setup performance, more specifically at the the size and compression of TLS certificates, due to the impact in the handshake phase. Finally, [46] evaluates the performance of several QUIC implementations over several emulated and real-world geostationary satellite links. the authors report poor performance for QUIC, specially when there is packet loss.

C. HTTP3 protocol

The Hypertext Transfer Protocol (HTTP) is used to access the vast majority of services on today's Internet. The protocol was born in the early 90s with the goal to allow multimedia content and hyper-textual document transfers over the Internet. HTTP/1.1 standardized version came out in 1997 [47]. In HTTP/1.1, only one resources can be in-flight on the underlying TCP connection, holding up all further resources behind it until it is fully downloaded. This is more generally known as Head-Of-Line (HOL) blocking. As the resources avalable in the Internet grew in size over the years, to achieve better page loading performance, Internet browsers started opening up several, up to six, parallel HTTP/1.1 connections per domain.

In 2014, HTTP's next version (HTTP/2) [55] was proposed with substantial changes in how data is framed and transported. As such, one of HTTP/2 main goals was to implement multiplexing of resources over a single underlying TCP connection. To achieve this goal, the protocol divides resource payloads into smaller uniquely-identified prefixed chunks, thus, allowing multiple resources on the wire.

Since 2022, HTTP/3 [56] is the most recent version of HTTP and it promises performance and security improvements

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2024.3383787

TABLE I: *Existing research on Encrypted DNS, QUIC and HTTP/3*

Protocols	Research Focus	References
Encrypted DNS	Performance comparison of DNS protocols	[33, 34, 37]
	Measurement on DNS adoption	[26, 35]
	DNS encryption and its performance	[13, 15, 36, 38]
	Security analysis of DNS protocols	[38, 39]
QUIC	Deployment and adoption	[40, 41]
	QUIC with IoT	[42]
	Performance of QUIC over different networks	[23, 43, 44]
	QUIC's performance over different workloads	[22]
	QUIC and TLS interplay	[45]
	QUIC over satellites	[46]
HTTP/3 (H3)	Resource Multiplexing	[48]
	HTTP Adaptive Streaming (HAS) over H3	[49]
	H3 with Lighthouse	[50]
	H3 adoption and performance measurement	[51–53]
	H3 over LEO satellites	[21]
	H3 with IoT	[54]

compared to HTTP/2. While HTTP/3 semantics and high-level features of HTTP/2 are kept intact, some core protocol aspects have been substantially reworked [57]. Beyond the replacement of the underlying transport protocol from TCP to QUIC, HTTP/3 comes with more efficient header compression, and advanced security features based on TLS 1.3.

Meanwhile, research (see: Table I) quantifying the benefits of HTTP3 in terms of Quality of Experience (QoE), HTTP features, different applications, and different network scenarios such as IoT, mobile and satellite networks has emerged with different outcomes: [48] compares resource multiplexing prioritization between HTTP2 and HTTP3 protocols. [49] investigates HTTP Adaptive Streaming (HAS) over HTTP3, proposing an optimization to the Adaptive Bitrate (ABR) algorithm using HTTP3 request cancellation, and [50] looks at diverse HTTP3 metrics with Lighthouse.

In [51] the authors run a measurement study looking at HTTP3 adoption and performance, where at the time it testified its benefits limited to a few scenarios with high latency or poor bandwidth. Later, [52] revisited the topic with a slight different outcome: While confirming that the benefits of HTTP3 were more visible in high latency scenarios and also mobile networks, the did not observe any improvements with video streaming. In [53], authors look at Quality of Experience (QoE) and the impact of local connectivity, server location and server software between HTTP2 and HTTP3, where they confirm better performance of HTTP3 over HTTP2 in challenging networking conditions.

Then, [21] looks at HTTP3 performance in Low-Earth Orbit (LEO) satellites with and without Performance Enhancing Proxies (PEP), where the authors indicate better HTTP3 performance with and without proxy compared to its predecessors. Finally, [54] looks at the MQTT IoT protocol over HTTP3 indicating that they could save one RTT to publish messages to the broker, which in typical high-latency or low-power IoT environments, is significant.

III. Methodology

To evaluate QUIC connection coalescing using DoQ + H3 $0-$RTT, our measurement setup (see: Fig. 2) 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 §VI).

The measurement scenario is web browsing where *Chromium* [58] 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 were downloaded on June, 2022 and are specifically 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. The websites are cloned and provided by *quic-go* H3 server with gzip compression for all data including html. 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 II 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 [59] is used to represent the fixed broadband scenarios, whereas the ERRANT dataset [60, 61] is used for mobile wireless access technologies. The delays and bandwidth are controlled using `netem`, where delay is always meant in the

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2024.3383787

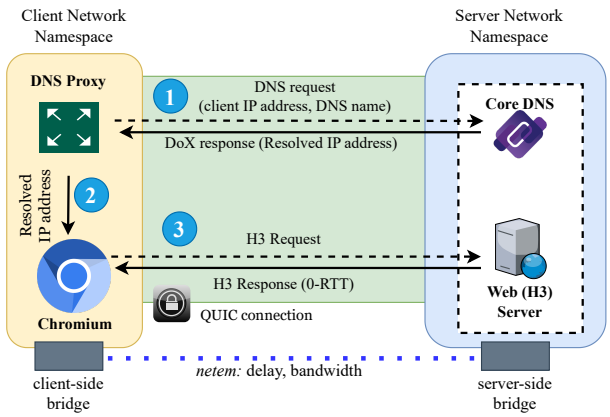


Fig. 2: *Measurement setup used to evaluate QUIC connection coalescing using DoQ + H3 0−RTT. The setup automates DNS resolution and Web browsing while emulating network conditions, such as delay and bandwidth of a user at the edge.*

sense of two-way delay, i.e., the round-trip time (RTT), where the on-way delay is assumed to be symmetrical.

To enable this setup, we used *Chromium 102.0.4991.0* and *CoreDNS 1.7.0* [62], where several changes were made to both these open source tools. *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. The used machine was an Ubuntu 18.04 with Kernel 5.4.0, featuring 2 Intel Xeon E5-2643 6-Core CPUs and 128GB of RAM. It is to be noted that during the evaluation, we did not find any limitation by the hardware.

IV. Evaluation

In order to evaluate QUIC connection coalescing, we first investigate the interaction of QUIC with DoQ and H3 in § IV-A, followed by an evaluation of the overhead of DoQ and DoH in comparison to the unencrypted DoUDP in § IV-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 § IV-C. Our goal is to observe how different access technologies influence the behavior of these protocols, but not to evaluate a representative mix of internet access connections. In our dataset, all these protocol combinations have a sample size of 57,436. 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.

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. 3 shows two relevant metrics for H3: *connect* duration (i.e., *connectEnd - connectStart*) and DoQ QUIC handshake duration measured in the DNS proxy. Here, *'connectStart'* signifies the timestamp immediately before the user starts establishing the connection to the server in order to retrieve the resource. In this experiment, the user establishes TCP and TLS sessions. On the other hand, *'connectEnd'* defines the timestamp immediately after the browser finishes establishing the connection to the server for retrieving the resource. 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. The `FIN bit (0x01)` of the frame type is set on frames that contain the final offset of the stream. Setting this bit indicates that the frame marks the end of the stream. Thus, 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 *connect* time is a valid metric to look at while measuring how long it takes until the first request is sent.

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

Access Technology	Delay (ms)	Download (Mbps)	Upload (Mbps)
Fibre	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

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. 4a and 4b reflect how the access technologies scale for fiber and 4G scenario respectively. It is observed from Fig. 4a 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

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2024.3383787

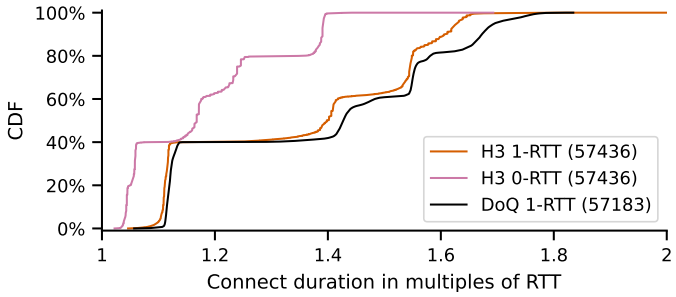
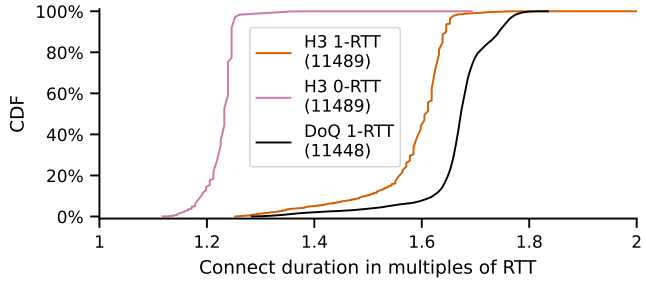
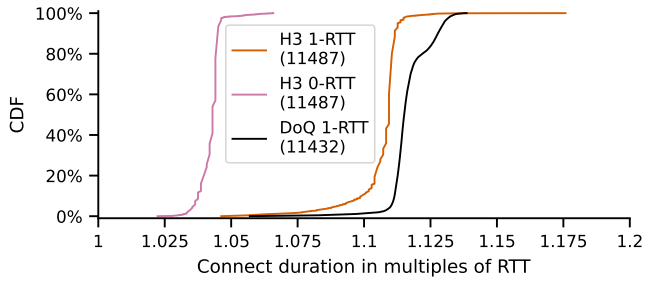


Fig. 3: *CDF of the QUIC handshake connect duration H3 for 1-RTT and 0-RTT, as well as DoQ 1-RTT for all scenarios. The values are normalized by the delay that was applied during the measurement to show how these metrics scale with round-trips.*



(a) Fiber scenario



(b) 4G Scenario

Fig. 4: *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.*

0.38 round-trips).

Comparing this observation to the difference in round-trips for 4G in Fig. 4b, we observe that the median for 1-RTT increases to 1.12 from 1.06 round-trips as for 0-RTT. The plot also shows how the different steps in Fig. 3 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. 4b. 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. 4b 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, as existing literature [25, 26, 46] confirms that the Address is already validated by receiving a 1-RTT packet; 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. 5 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 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. 6a and 6b 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

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2024.3383787

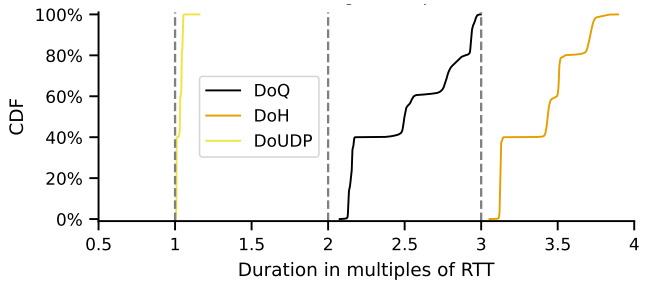


Fig. 5: *CDF of DNS exchange duration in multiples of round trip times for all scenarios. Only DoUDP scales with the number of expected round-trips.*

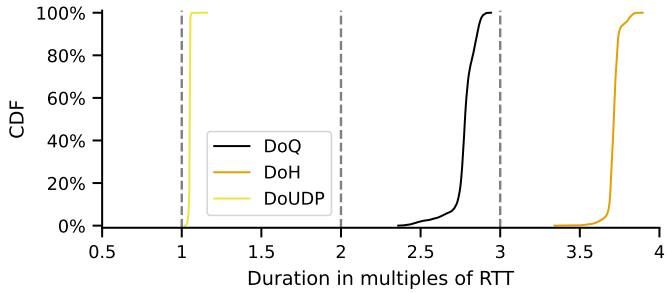
round-trip to 3.05 and 3.12 round-trips. This shows that the range of values for 4G is much smaller, meaning there is less variation in the data and there is no long tail as well. Analysing other access technology scenarios, the left tail appears to be the largest for fiber whereas it gets smaller when looking at scenarios with higher delay.

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.

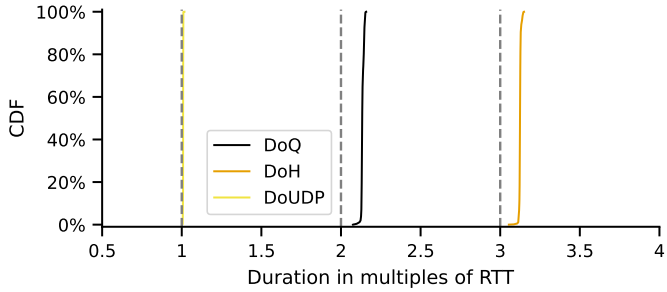
Digging deeper into this aspect, the measurement also contains data for the RTT of TCP (i.e., client sends a SYN and server responds with a SYN-ACK). The TCP round-trip times are analyzed to inspect whether the reason for the unusual scaling of DoH is rooted in something related to the TCP handshake 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. 7a 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. 7b. 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.



(a) Fiber Scenario



(b) 4G Scenario

Fig. 6: *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.*

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.

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

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

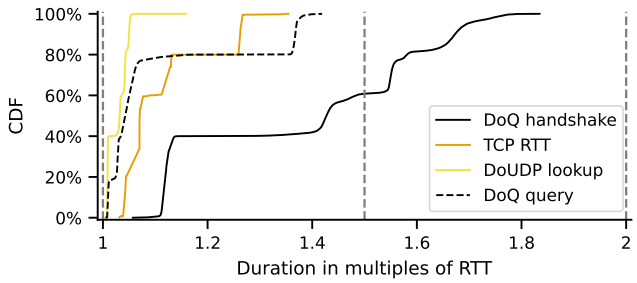
This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2024.3383787

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 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.

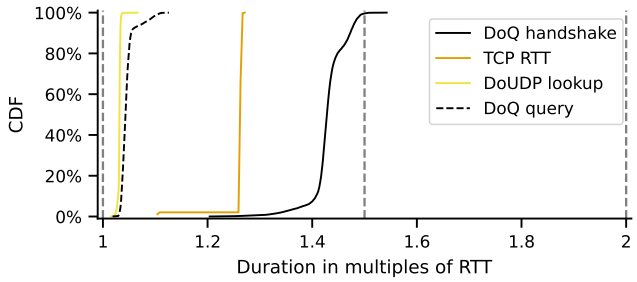
The first set of experiment provides an overview of the median PLT increase for all the considered access technologies and web pages. The relative increase over the DoUDP + H3 $1-$RTT baseline is calculated for three protocol combinations: QUIC connection coalescing, DoQ and DoH. The relative increase 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. 8 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*. 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. 9. Note that in theory, the relative increase can be calculated using the value of the



(a) All Scenarios



(b) Cable Scenario

Fig. 7: *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.*

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.

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

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2024.3383787

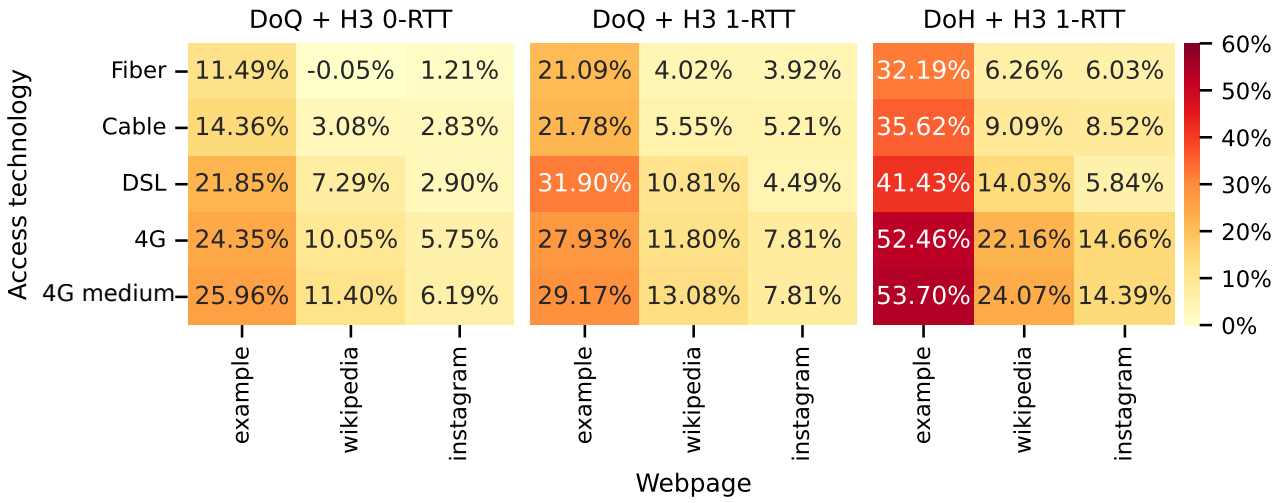


Fig. 8: *Heat map of relative PLT increase over DoUDP baseline for QUIC connection coalescing (i.e. DoQ + H3* $0-$*RTT), DoQ + H3* $1-$*RTT, and DoH + H3* $1-$*RTT. Cost of encryption is substantially reduced when encrypted connections are coalesced using DoQ + H3* $0-$*RTT.*

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. 9 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 § IV-B). However, the difference between H3 $0-$RTT and $1-$RTT does not, as can be seen in Fig. 9 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.

> ***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.*

D. On a Deep Dive into Website Categories

The effects of both the dimensions (i.e. access technology and website complexity) on web performance and its scaling with respect to the underlying protocols is studied here in terms of absolute PLT values. Owing to the simplicity of the HTML page (*example page*), the observed PLT distributions in the previous section are justified, hence further discussion about it is omitted. For the remaining two categories of websites, the optimal scenarios for fixed broadband (fiber) and cellular connectivity (4G with good reception) are analyzed.

HTML Page with Javascript: The *Wikipedia page* consists of an index HTML document (18,252 bytes), two *png* logos (15,857 and 2,039 Bytes), an *svg* sprite (17,229 Bytes) and a Javascript resource (614 Bytes, no compression). These byte values are not an exact match with the original Wikipedia Page.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2024.3383787

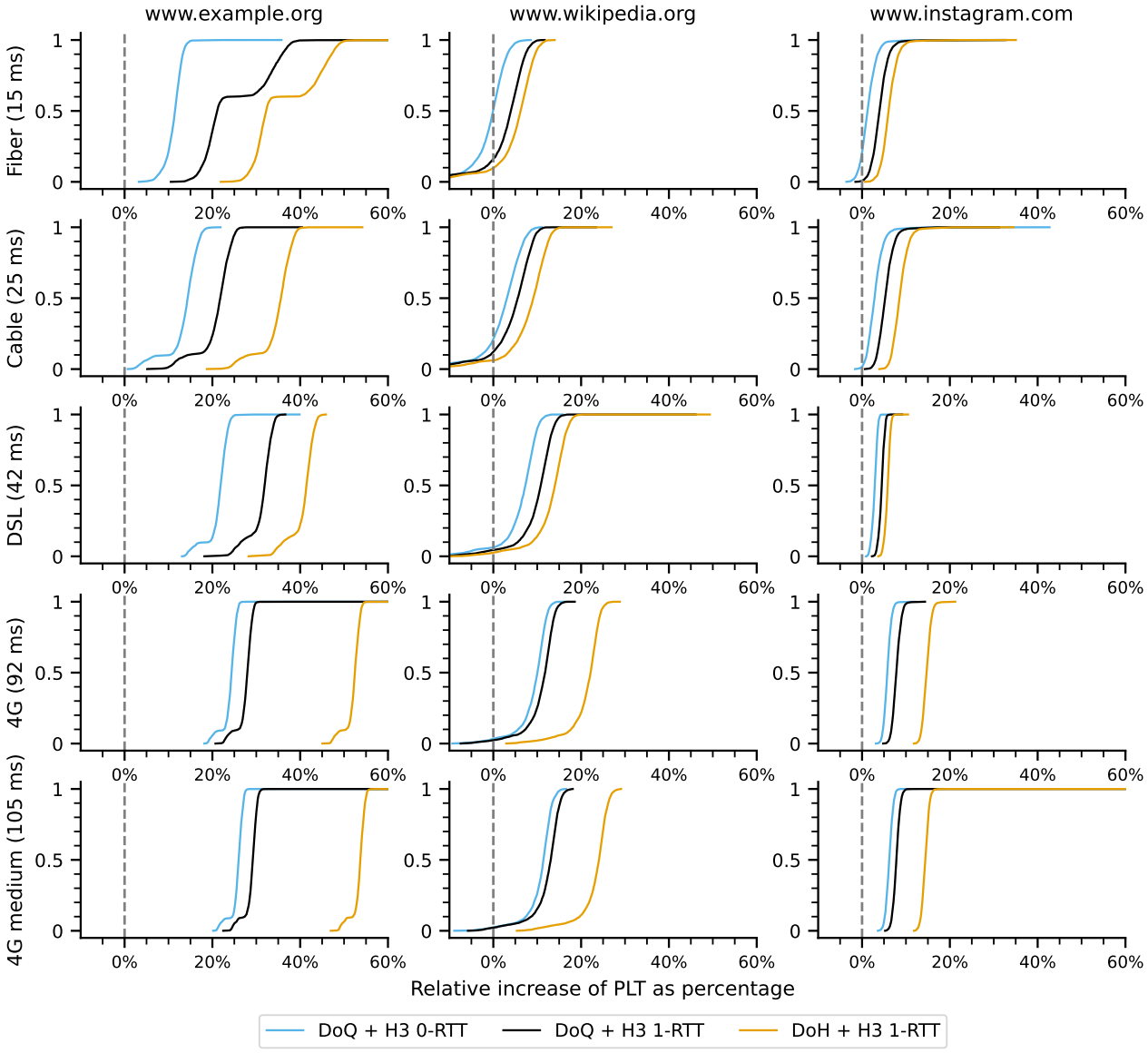


Fig. 9: *Grid of CDFs showing the relative increase of QUIC connection coalescing (i.e. DoQ + H3* $0-$*RTT), DoQ + H3* $1-$*RTT, and DoH + H3* $1-$*RTT over the horizontal DoUDP baseline for the five access technologies and the three web pages. Relative changes between the protocol combinations are affected by both of these dimensions.*

Looking at a common web browsing scenario, the median is fixed to 630.4 ms which acts as the baseline for the measurement. This is also where the simulated QUIC connection coalescing setup (median 630.0 ms) matches perfectly with the baseline. Fig. 10a shows the PLTs for the fiber scenario (RTT 14.8 ms). As all the protocol combinations have a long left tail, only the median values are discussed here. It appears to be equal to the baseline across all percentiles, indicating that this is not just an artifact of the data: the baseline indeed has a P25 (Q1) value of 615.5 ms and a P75 (Q3) value of 642.1 ms, resulting in an interquartile range (IQR) of 26.6 ms. For QUIC connection coalescing, Q1 is observed to be 616.4 ms and Q3 is 641.3 ms with an IQR of 24.9 ms. The DoQ with HTTP/3 1-RTT setup has a median of 655.7 ms, which increases to 669.8 ms on changing the DNS protocol to DoH. This results in a difference of 14.1 ms between the two setups, which is almost exactly the round-trip time. This is lower than the benefit of QUIC connection coalescing over DoQ (median difference of 26 ms), which also means that the optimized QUIC setup saves more than a single round-trip time.

Figure 10b depicts the results for the same website under the 4G scenario. In this case, the DNS protocol being used has some influence on the PLT, unlike, the scenario where HTTP/3 0-RTT is used which produces lesser benefits due to the exchange of settings values. The observed median difference between 0-RTT and 1-RTT is on average 15.4 ms, which is lower than the one seen for 0-RTT fiber scenario (median difference of 26 ms). This means that benefit obtained from QUIC connection coalescing (based on HTTP/3 0-RTT) is somewhat dependant on the round-trip delay, however, the major component is still lower processing delay. The median for the QUIC connection coalescing setup is 965.0 ms while

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2024.3383787

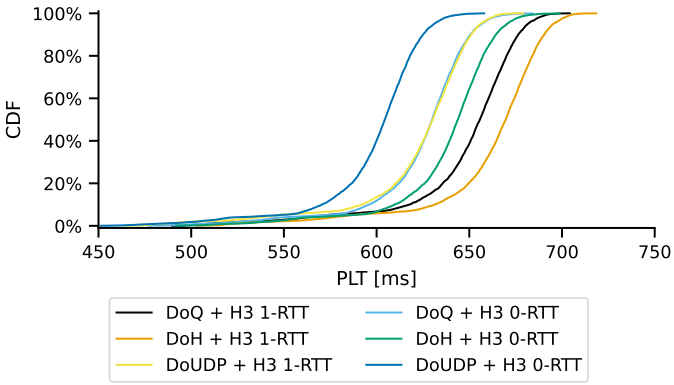
that of the baseline setup is 876.9 ms, thus, bearing a difference of 88.1 ms. Correspondingly, the DoQ setup has a median of 980.4 ms, thereby, having a difference of 103.5 ms to the baseline. Finally, the difference is median values between the DoQ and DoH setups is 90.8 ms which is highly reflected in the round-trip time (91.9 ms) for the setup as well. Overall, this means that while the QUIC connection coalescing setup does perform fairly well (excluding the baseline), maximum benefit can be gained from using DoQ over DoH.

HTML Page with Javascript and CSS: A major part of rendering the *Instagram page* is related to building the user interface from seven Javascript resources, two style sheets, a cookie popup banner and a login form. The various images (such as the Instagram, App Store and Play store logos) embedded in the *Instagram page*, are triggered by scripts loaded after the index HTML document is fetched. The website also attempts to load app screenshots, but they are never rendered in the current setup due to the viewport size being too small. However, an image of a smartphone that acts as a border around these screenshots is mirrored correctly but isn't fetched. Due to these reasons, the measured PLTs are different from that of the real website.

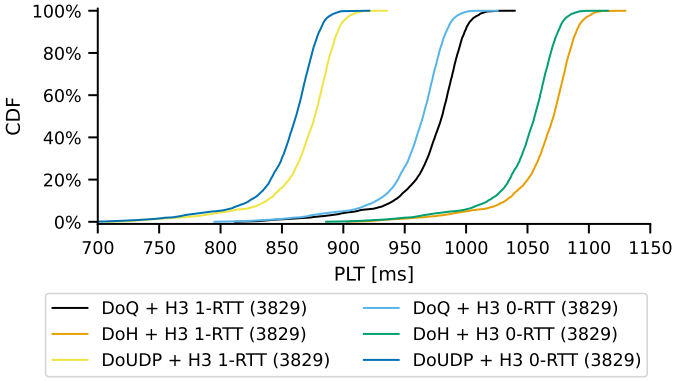
Figure 11a depicts the PLTs for the fiber scenario. It is observed that PLTs of the *Instagram page* is higher compared to the *Wikipedia page* due to its greater complexity. We also observe that difference between 0-RTT and 1-RTT is relatively closer to the differences between the DNS protocols. Here, 0-RTT saves on average 17.6 ms which is slightly more than one round-trip time (14.8 ms). However, *Instagram page* achieves lower savings compared to that of the *Wikipedia page* (26 ms). This implies that benefits of QUIC connection coalescing also depend on the website's complexity. The median for the baseline is 651.4 ms while the median for the QUIC connection coalescing setup is 659.3 ms thus bearing a difference of 7.9 ms. Similarly, by comparing the DoQ setup to the baseline, we observe a difference of 25.5 ms with the median at 676.9. The difference between DoQ and DoH (690.7 ms median) is 13.8 ms which is 1 ms lower than the applied delay. Similar to the *Wikipedia page*, the potential benefit of using a QUIC connection coalescing setup is greater than the benefit of using DoQ over DoH for encrypted DNS.

Lastly, Figure 11b shows the PLTs for the 4G scenario. Quite similar to the *Wikipedia page* the difference between medians of DNS protocols now outweighs the difference between 0-RTT and 1-RTT which implies the potential benefits of QUIC connection coalescing over DoQ. The former is 90.9 ms when going from DoH to DoQ while the latter is 27.4 ms on average, which is more than the fiber scenario. This is just the reverse of the effect seen with the *Wikipedia page* where the fiber scenario had more PLT reductions in QUIC connection coalescing than 4G. The difference between the two encrypted DNS protocols is close to the delay on the connection (91.9 ms). This effect is visible across websites and access technologies which is quite obvious as DNS timings are independent from the websites being measured.

In conclusion, using the QUIC 0-RTT handshake does not result in a speedup of one round-trip with regard to the browser sending the initial *GET* request. While using 0-RTT saves some milliseconds of processing delay in the PLTs, it is however reflected differently depending on the website and access technology. For the *Wikipedia page* under the fiber scenario, QUIC connection coalescing reduces median PLT by 26 ms compared to DoQ, which decreases to 15.4 ms under 4G. For the *Instagram page* on the other hand, QUIC connection coalescing saves 17.6 ms when simulating a fiber connection, which increases to 27.4 ms for 4G. Furthermore, the difference between DoUDP and DoQ appears to be higher than one round-trip, while the difference between DoQ and DoH is very close to one round-trip. This effectively means that the QUIC connection coalescing setup has the highest impact on connections with low delay. This is because in our measurement setup it does not scale with RTT in an obvious way. Furthermore, specific DNS protocols are less important for more complex websites (assuming all resources are served by the same host), since the browser rendering takes more time and thus DNS performance has less impact.



(a) Fiber Scenario



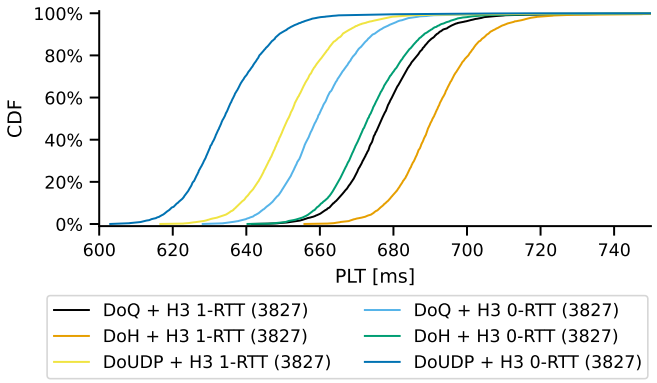
(b) 4G Scenario

Fig. 10: *CDF of the PLTs for all protocol combinations for the Wikipedia page.*

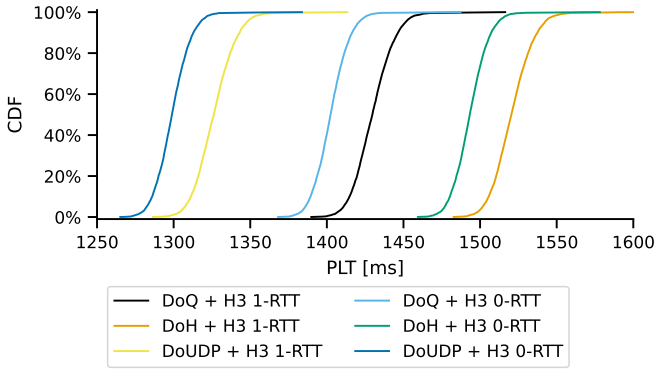
Takeaway: *Overall, our findings show that QUIC connection coalescing is the best option for encrypted communication on the Internet, however it is more beneficial for less complex websites. Also, the performance gains vary depending on the website and access technology combination. Lastly, QUIC connection coalescing setup has the highest impact on connections with low delay.*

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2024.3383787



(a) Fiber Scenario



(b) 4G Scenario

Fig. 11: *CDF of the PLTs for all protocol combinations for the Instagram page.*

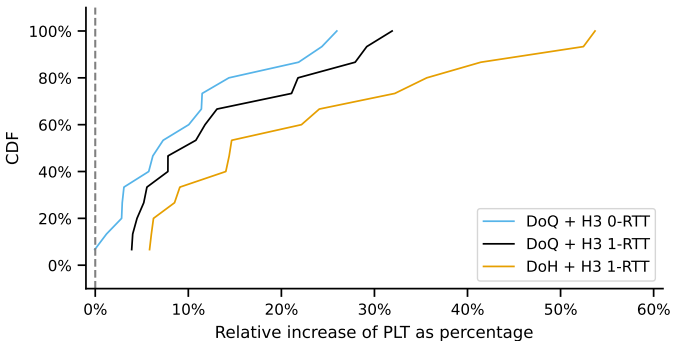


Fig. 12: *CDF of the relative increase of protocol combinations over the DoUDP baseline.*

Summary

To summarize, Fig. 12 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 already explained, 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. While we agree that in some use cases with browsers, e.g. static connection and same destination (hostname), the benefit of 0-RTT could be limited to the first visited website, several other realistic scenarios, e.g., end users changing the access network, e.g., moving from mobile to WiFi and vice-versa in a smartphone, closing the application (browser) or reaching a different Point-Of-Presence of a content provider, could benefit from having 0-RTT.

V. Discussion

There can be additional challenges, when QUIC is used to coalesce name resolution via DoQ and Web content delivery via H3 with 0-RTT over the same edge server. Next, we would like to comment on some aspects when it comes to performance and privacy in the current state of the Internet and for its future, with an eventual adherence to coalescence of DNS, QUIC and H3, specially.

A. Connection Coalescence in the Internet

Popular Internet content providers such as CDNs nowadays host DNS services, even their own DNS as a service, to offer content from third-party customers. From a CDNs' perspective the clear benefit of connection coalescence is the drastic reduction in the number of parallel connections needed by a web browser to locate (DNS) and fetch (website) content arriving at their infrastructure. For instance, to read the content of a simple blog entry today, this could easily involve tens of DNS requests to resolve another tens of hostnames, serving (sub-)subresources to construct the website as part of the operation. Again, from content serving part of the network, this means a significant load per client arriving at the servers and potentially causing scalability issues. On the other hand, one can argue that in all the involved requests that much of the end user's metadata in this operation, e.g., via the absence of an encrypted Server Name Indicator (SNI), is clearly exposed to the network (not only the CDN). With the SNI, an on-path eavesdropper could easily fingerprint the traffic coming from the end user and determine the interactions on the website. From the end user perspective coalescing the connections into

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2024.3383787

a single means placing a strong level of trust in the CDN, which will have access to all data from the DNS requests to the website(s) interactions. The idea of coalescing connections is not new, it was brought from earlier HTTP versions, and most modern browsers already reuse open connections to the same website, assuming the connection identification and certificate are the same. For browsers, the clear benefit is to reduce the large number of parallel connections that can be open and connection management. Also, different forms of scheduling requests and parallelization routines can be done at the browser since there is better visibility in how content is being delivered.

From our measurements, the best option for end users today if connection coalescence was already a reality, is the reduction of necessary RTTs from the DNS request to the content retrieval. However, QUIC has matured over the years to show not only performance but flexibility benefits as a transport layer protocol to encrypt Internet communication. We provide a different example, where QUIC can be used to address concerns around privacy of users in general: Many Internet players actually need some information exposure from protocols to perform tasks such as network management. Think, for instance, of an ISP that needs some understanding about a sudden traffic influx: Is it malicious or an unexpected user activity routed to a different network segment due to an outage? With QUIC, a good portion of this traffic is invisible to the ISP that is not able to decrypt the end-to-end connection. To address such concerns, often called "pervasive encryption", there are solutions such as Multiplexed Application Substrate over QUIC Encryption (MASQUE), which proposes QUIC as a substrate to tunnel any type of traffic. Beyond this initial goal, it also attempts to create a collaborative approach, e.g., via a MASQUE proxy, where pieces of the end-to-end communication can be selectively exposed to certain parties, i.e., the ISP, along the end-to-end path. Beyond the demonstrated benefits of QUIC's built-in features to carry multiple streams in a single and secure connection, it is flexible enough to also address other privacy concerns.

Both DNS and HTTP allow content to be cached, where such cache nodes are currently hosted separately. Our architectural design proposes to leverage these cache nodes to offer both DNS and Web services together. The paper demonstrates the performance and privacy benefits that can be realized with such a deployment. To promote this idea further, we are actively engaging with the operations (RIPE) and standards (IETF) communities to bring this design to deployment.

B. Web Security, User Privacy and Trust

While the trend of secure encrypted communication in the Internet with TLS is overall positive, the encryption of DNS requests has been heavily debated due to the privacy implications side effects imposed to end users: The DNS infrastructure has long been centralized and, by enabling encryption with approaches discussed in this paper such as DoH or DoQ and DoT, it delivers all user data in the hand of a few Internet players, i.e., hyper-giants. Although the encryption of DNS requests is unquestionably positive, it has clear consequences to performance, privacy, competition, and availability of DNS. A growing concern is the difficulty in the control and choice of the DNS recursive resolver. For instance, in a desperate attempt to lose DNS queries' visibility, it is known that ISPs partnered with web browsers to become trusted DNS resolvers. There are a few solutions from distributing DNS queries in different ways, e.g., hash-based or randomly, to different recursive resolvers to run a proxy under control of the end user, where they can configure the "visible" parts of their DNS requests to the outside Internet.

With the deployment of connection coalescing, an increasing amount of user data will be delivered to large content-delivery hyper-giants. As such, the system can still be viewed as a trade-off between performance and privacy, depending on the sensitivity of the user, as connection coalescing can lead to centralization of trust [63]. With the EU, via the Digital Services Act, working towards ensuring that hyper-giants adhere to stricter privacy and transparency obligations, the concept of Web privacy within the context of hyper-giants will continue to evolve in the coming years. Along the lines with ISPs partnering with web browsers to become their trusted DNS recursive resolver, the same ISPs or other networks used more as transport networks, i.e., shuffling traffic, such as mobile operators, can directly collaborate with the hypergiants so that portions of the traffic stays within the ISPs. This collaborative approach is definitely more promising to improve privacy for Internet users rather than circumventing encryption or blocking traffic.

While, the presented method prevents an intruder from plain eavesdropping the browsing behaviour of the clients and/or launching man-in-the-middle attacks, whether pattern inference of encrypted packets using machine learning methods can reveal parts of browsing behavior such as with website fingerprinting (WFP) attacks [64] is an area of further exploration. It is technically possible to censor HTTP traffic, since the SNI option in TLS is not encrypted and still visible in H3. This could be one approach to identify and deliberately block QUIC traffic. The built-in encryption of QUIC makes it less vulnerable to other types of censorship techniques such as connection tracking, since the connection identification values change during the time of the connection. Also, even when IP addresses may change, the same QUIC connection can continue to exist.

Within this context, it would further be interesting to investigate how the performance varies in the presence of background traffic coupled with a WFP attack. Further, the presented method does not yet prevent the association of a user (via source IP address) from the requested content (via destination IP address). As such, combining the method with private relays, such as MASQUE, that leverage QUIC as the underlying protocol and is currently under standardization at the IETF, would be an interesting new direction to tighten the privacy properties of the system.

VI. 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.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2024.3383787

As a consequence, the performance metrics are computed by summing DNS time and HTTP time. Taking this factor into account, the evaluation shows a lower bound of the possible performance benefits of coalescing. 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 [65]. Cloudflare has also added DoH3 support to their public DNS service in March 2022 [66]. Hence, we plan to extend our work with DoH3 further by blurring the boundaries between DNS resolution and Web content delivery.

VII. 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. [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. [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. [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. [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. [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. [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. [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. [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. [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

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2024.3383787

[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, 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

[26] 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

[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] D. Naylor, A. Finamore, I. Leontiadis, Y. Grunenberger, M. Mellia, M. Munafò, K. Papagiannaki, and P. Steenkiste, "The Cost of the "S" in HTTPS," in *Proceedings of the 10th ACM International on Conference on Emerging Networking Experiments and Technologies*, 2014, p. 133–140.

[29] J. Sengupta, M. Kosek, J. Fries, S. Ferlin-Reiter, and V. Bajpai, "On Cross-Layer Interactions of QUIC, Encrypted DNS and HTTP/3: Design, Evaluation and Dataset," 2024. [Online]. Available: https://github.com/Sree2021/TNSM-2024-Web_Privacy

[30] J. Sengupta, M. Kosek, J. Fries, P. Dikshit, and V. Bajpai, "Web Privacy By Design: Evaluating Cross-layer Interactions of QUIC, DNS and H/3," in *2023 IFIP Networking Conference (IFIP Networking)*, 2023, pp. 1–9. [Online]. Available: https://doi.org/10.23919/IFIPNetworking57963.2023.10186362

[31] P. V. Mockapetris, "Domain names - concepts and facilities," *RFC*, vol. 1034, pp. 1–55, 1987. [Online]. Available: https://doi.org/10.17487/RFC1034

[32] "Domain names - implementation and specification," RFC 1035, Nov. 1987. [Online]. Available: https://www.rfc-editor.org/info/rfc1035

[33] G. C. M. Moura, M. Müller, M. Davids, M. Wullink, and C. Hesselman, "Fragmentation, Truncation, and Timeouts: Are Large DNS Messages Falling to Bits?" in *Passive and Active Measurement*. Springer International Publishing, 2021, pp. 460–477. [Online]. Available: https://doi.org/10.1007/978-3-030-72582-2_27

[34] M. Kosek *et al.*, "Measuring DNS over TCP in the Era of Increasing DNS Response Sizes: A View from the Edge," *SIGCOMM CCR*, vol. 52, no. 2, pp. 44–55, June 2022. [Online]. Available: https://doi.org/10.1145/3544912.3544918

[35] S. García, K. Hynek, D. Vekshin, T. Cejka, and A. Wasicek, "Large Scale Measurement on the Adoption of Encrypted DNS," *CoRR*, vol. abs/2107.04436, 2021. [Online]. Available: https://arxiv.org/abs/2107.04436

[36] G. Kambourakis and G. Karopoulos, "Encrypted dns: The good, the bad and the moot," *Computer Fraud & Security*, vol. 2022, no. 5, 2022. [Online]. Available: https://doi.org/10.12968/S1361-3723(22)70572-6

[37] J. Jung, E. Sit, H. Balakrishnan, and R. T. Morris, "DNS performance and the effectiveness of caching," *IEEE/ACM Trans. Netw.*, vol. 10, no. 5, pp. 589–603, 2002. [Online]. Available: https://doi.org/10.1109/TNET.2002.803905

[38] M. Lyu, H. H. Gharakheili, and V. Sivaraman, "A survey on dns encryption: Current development, malware misuse, and inference techniques," *ACM Comput. Surv.*, dec 2022. [Online]. Available: https://doi.org/10.1145/3547331

[39] R. Houser, Z. Li, C. Cotton, and H. Wang, "An Investigation on Information Leakage of DNS over TLS," in *Proceedings of the 15th International Conference on Emerging Networking Experiments And Technologies*, ser. CoNEXT, 2019, p. 123–137. [Online]. Available: https://doi.org/10.1145/3359989.3365429

[40] S. Cook *et al.*, "QUIC: Better for what and for whom?" in *ICC*, 2017. [Online]. Available: https://doi.org/10.1109/ICC.2017.7997281

[41] J. Rüth, I. Poese, C. Dietzel, and O. Hohlfeld, "A First Look at QUIC in the Wild," in *Passive and Active Measurement*. Springer International Publishing, 2018, pp. 255–268. [Online]. Available: https://doi.org/10.1007/978-3-319-76481-8_19

[42] F. Fernández, M. Zverev, P. Garrido, J. R. Juárez, J. Bilbao, and R. Agüero, "And QUIC meets iot: performance assessment of MQTT over QUIC," in *16th International Conference on Wireless and Mobile Computing, Networking and Communications, WiMob*, 2020, pp. 1–6. [Online]. Available: https://doi.org/10.1109/WiMob50308.2020.9253384

[43] G. K. Choudhary *et al.*, "Novel multipipe QUIC protocols to enhance the wireless network performance," in *2020 IEEE Wireless Communications and Networking Conference, WCNC*. IEEE, 2020, pp. 1–7. [Online]. Available: https://doi.org/10.1109/WCNC45663.2020.9120821

[44] J. Manzoor, L. Cerdà-Alabern, R. Sadre, and I. Drago, "On the Performance of QUIC over Wireless Mesh Networks," *Journal of Network and Systems Management*, vol. 28, no. 4, p. 1872–1901, Oct 2020. [Online]. Available: https://doi.org/10.1007/s10922-020-09563-8

[45] M. Nawrocki, P. F. Tehrani, R. Hiesgen, J. Mücke, T. C. Schmidt, and M. Wählisch, "On the interplay between tls certificates and quic performance," in *Proceedings of the 18th International Conference on Emerging Networking EXperiments and Technologies*, ser. CoNEXT. Association for Computing Machinery, 2022, p. 204–213. [Online]. Available: https://doi.org/10.1145/3555050.3569123

[46] S. Endres, J. Deutschmann, K. Hielscher, and R. German, "Performance of QUIC implementations over geostationary satellite links," 2022. [Online]. Available: https://arxiv.org/abs/2202.08228

[47] H. Nielsen, J. Mogul, L. M. Masinter, R. T. Fielding, J. Gettys, P. J. Leach, and T. Berners-Lee, "Hypertext Transfer Protocol – HTTP/1.1," RFC 2616, Jun. 1999. [Online]. Available: https://rfc-editor.org/rfc/rfc2616.txt

[48] R. Marx, T. D. Decker, P. Quax, and W. Lamotte, "Resource multiplexing and prioritization in HTTP/2 over TCP versus HTTP/3 over QUIC," in *Web Information Systems and Technologies WEBIST*, vol. 399, 2019, pp. 96–126. [Online]. Available: https://doi.org/10.1007/978-3-030-61750-9_5

[49] D. Lorenzi, M. Nguyen, F. Tashtarian, S. Milani, H. Hellwagner, and C. Timmerer, "Days of future past: an optimization-based adaptive bitrate algorithm over HTTP/3," in *EPIQ '21: Proceedings of the 2021 Workshop on Evolution, Performance and Interoperability of QUIC*, 2021, pp. 8–14. [Online]. Available: https://doi.org/10.1145/3488660.3493802

[50] D. Saif *et al.*, "An Early Benchmark of Quality of Experience Between HTTP/2 and HTTP/3 using Lighthouse," *ICC*, 2021. [Online]. Available: https://doi.org/10.1109/ICC42927.2021.9500258

[51] M. Trevisan, D. Giordano, I. Drago, and A. S. Khatouni, "Measuring HTTP/3: Adoption and Performance," in *19th Mediterranean Communication and Computer Networking Conference, MedComNet*, 2021, pp. 1–8. [Online]. Available: https://doi.org/10.1109/MedComNet52149.2021.9501274

[52] G. Perna, M. Trevisan, D. Giordano, and I. Drago, "A first look at HTTP/3 adoption and performance," *Computer Communications*, vol. 187, pp. 115–124, 2022. [Online]. Available: https://doi.org/10.1016/j.comcom.2022.02.005

[53] A. Gupta and R. Bartos, "User Experience Evaluation of HTTP/3 in Real-World Deployment Scenarios," in *25th Conference on Innovation in Clouds, Internet and Networks (ICIN)*, 2022, pp. 17–23. [Online]. Available: https://doi.org/10.1109/ICIN53892.2022.9758130

[54] D. Saif and A. Matrawy, "APure HTTP/3 Alternative to MQTT-over-QUIC in Resource-Constrained IoT," in *2021 IEEE Conference on Standards for Communications and Networking, CSCN*, 2021, pp. 36–39. [Online]. Available: https://doi.org/10.1109/CSCN53733.2021.9686113

[55] M. Belshe, R. Peon, and M. Thomson, "Hypertext Transfer Protocol Version 2 (HTTP/2)," RFC 7540, May 2015. [Online]. Available: https://www.rfc-editor.org/info/rfc7540

[56] M. Bishop, "HTTP/3," RFC 9114, Jun. 2022. [Online]. Available: https://www.rfc-editor.org/info/rfc9114

[57] R. Marx, J. Herbots, W. Lamotte, and P. Quax, "Same Standards, Different Decisions: A Study of QUIC and HTTP/3 Implementation Diversity," in *Workshop on the Evolution, Performance, and Interoperability of QUIC*, ser. EPIQ, 2020, pp. 14–20. [Online]. Available: https://doi.org/10.1145/3405796.3405828

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

[59] 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

[60] 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

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

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

[63] T. V. Doan, R. van Rijswijk-Deij, O. Hohlfeld, and V. Bajpai, "An empirical view on consolidation of the web," *ACM Trans. Internet Techn.*, vol. 22, no. 3, pp. 70:1–70:30, 2022. [Online]. Available: https://doi.org/10.1145/3503158

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2024.3383787

[64] S. Siby, M. Juárez, C. Díaz, N. Vallina-Rodriguez, and C. Troncoso, "Encrypted DNS -> Privacy? A Traffic Analysis Perspective," in *27th Annual Network and Distributed System Security Symposium, NDSS*. The Internet Society, 2020. [Online]. Available: https://www.ndss-symposium.org/ndss-paper/encrypted-dns-privacy-a-traffic-analysis-perspective/

[65] 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

[66] 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/

Jayasree Sengupta is currently working as a Postdoctoral Researcher at CISPA Helmholtz Center for Information Security, Germany. She received her PhD (2022) in Computer Science from the Indian Institute of Engineering Science and Technology, Shibpur, India, and MTech (2017) in Distributed and Mobile Computing from Jadavpur University, India. She is a recipient of the IFIP Networking Best Paper Award (2023) and JNCA Best Survey Paper Award (2022). Her research interests include Applied Cryptography, Blockchains, Network Security, and Data Privacy.

Mike Kosek is a Researcher at the Chair of Connected Mobility at the Technical University of Munich, Germany. His research focuses on Internet measurements in general, and transport protocol standardization, development, and deployment, in particular.

Justus Fries is a Researcher at the Chair of Connected Mobility at Technical University of Munich, Germany. His research interests include Internet measurements, transport protocols and the Web.

Simone Ferlin-Reiter (PhD, 2017) is Senior Performance Engineer at Red Hat AB and Senior Adjunct Lecturer at Karlstad University, Sweden. She received her Dipl.-Ing. with a major in Telecommunications from Friedrich-Alexander Erlangen-Nürnberg University, Germany in 2010 and her Ph.D. in Computer Science from the University of Oslo, Norway in 2017. Her interests lie at the intersection of cellular networks and the Internet, and her research focuses on computer networks, transport protocols, congestion control, network and system performance, security, and measurement. Her PhD thesis focused on improving robustness in multipath transport for heterogeneous networks using MPTCP, resulting in several upstream contributions. She is actively involved in the technical committees of major conferences and journals in these areas.

Vaibhav Bajpai is Professor and Head of the data-intensive Internet computing research group at the Hasso Plattner Institute and the University of Potsdam. Previously, he was an independent research group leader at the CISPA Helmholtz Center for Information Security, Hannover. Before that, he was a senior researcher at the Department of Computer Science at the Technical University of Munich. He received his PhD (2016) and Masters (2012) degrees from Jacobs University Bremen. His research focuses on improving Internet operations using data-intensive methods and by building real-world systems and models.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/