Measuring the Performance of iCloud Private Relay

Trevisan Martino, Idilio Drago, Paul Schmitt, Francesco Bronzino

Measuring the Performance of iCloud Private Relay

Martino Trevisan[†], Idilio Drago[‡], Paul Schmitt[⋆], Francesco Bronzino[⋆⋆]

[†]University of Trieste, [‡]University of Turin, [⋆]University of Hawaii,
[⋆⋆]Univ Lyon, EnsL, UCBL, CNRS, LIP

Abstract. Recent developments in Internet protocols and services aim to provide enhanced security and privacy for users' traffic. Apple's iCloud Private Relay is a premier example of this trend, introducing a well-provisioned, multi-hop architecture to protect the privacy of users' traffic while minimizing the traditional drawbacks of additional network hops (e.g., latency). Announced in 2021, the service is currently in the beta stage, offering an easy and cheap privacy-enhancing alternative directly integrated into Apple's operating systems. This seamless integration makes a future massive adoption of the technology very likely, calling for studies on its impact on the Internet. Indeed, the iCloud Private Relay architecture inherently introduces computational and routing overheads, possibly hampering performance. In this work, we study the service from a performance perspective, across a variety of scenarios and locations. We show that iCloud Private Relay not only reduces speed test performance (up to 10x decrease) but also negatively affects page load time and download/upload throughput in different scenarios. Interestingly, we find that the overlay routing introduced by the service may increase performance in some cases. Our results call for further investigations into the effects of a large-scale deployment of similar multi-hop privacy-enhancing architectures. For increasing the impact of our work we contribute our software and measurements to the community.

1 Introduction

The privacy of Internet users has become one of the most discussed issues in the field of networking. New protocols and services are being developed with strong privacy guarantees, while privacy-enhancing technologies are opening opportunities for new markets. iCloud Private Relay (PR) is a new service recently created by Apple that is integrated into the company's operating systems (i.e., MacOS, iOS, iPadOS). Initially launched in 2021, it offers users the possibility of forwarding traffic via a multi-party relay [19], offering a service that in many ways resembles a VPN but differs in privacy guarantees. The architecture results in no party (neither Apple nor their infrastructure partners) holding *both* user identity and the contacted servers, whereas a VPN architecture simply shifts trust to the VPN which has access to both. The seamless integration of the service in the Apple OSes, its low cost ($0.99 per month for the cheapest plan) and

its low entry barrier suggest that a large adoption of the service is very likely, with an anticipated major impact on Internet traffic [16] moving forward.

iCloud Private Relay works with a multi-party relay architecture: The client operating system connects to an ingress proxy (operated by Apple) using an encrypted connection over QUIC [4]. The ingress proxy routes the client traffic to an egress proxy (currently operated by one of Akamai, Cloudfare, and Fastly) that forwards the traffic to the destination server requested by the user. With this architecture, the ingress and egress proxies can only see the client's or the server's IP address, respectively, but never both. Equally, eavesdroppers (e.g., ISPs) can observe the traffic of multiple users to/from ingress and egress proxies and thus cannot easily profile individual users' activity from the traffic [26].

The possibility of a major adoption of the service in the short term raises questions about its impact on the internet. Similar privacy protection mechanisms, such as VPNs, onion routing [24] and Tor [10] have been studied in terms of both performance and privacy [13,2]. For example, the authors of [6] uncover the websites a user is visiting when connected via Tor by relying on side channels such as packet sizes and timing. Similarly, multiple authors [15,2] have studied the impact of privacy-enhancing technologies on Internet performance. For PR, however, we are aware of a single study focusing on the service [16], which focused on describing the system architecture and its deployment footprint, neglecting implications on performance and user-perceived quality of experience.

In this work, we focus on the impact of iCloud Private Relay on web performance. We set up active experiments using Apple devices and design multiple workloads to assess the effects of PR on different scenarios. We deploy our testbed across multiple locations and gather several metrics associated with users' Quality of Experience (QoE), such as page load time, throughput, and latency. Apple notes [3] that iCloud Private Relay can negatively affect web speed tests as such tests routinely use "several simultaneous connections to deliver the highest possible result", but goes on to claim that "actual browsing experience remains fast." Therefore we design our experiments to assess these claims, including speed tests and web browsing with and without PR in place.

Our results show that iCloud Private Relay does impact performance. We confirm a significant reduction in the throughput measured with speed tests, e.g., with up to 10-fold slower download throughput when using PR. We notice a performance penalty in web browsing too, observing a 60% increase in page load time in some cases. Performance impairments also occur in cases where a single connection is used to download a large file, thus questioning the claim that several simultaneous connections are the root cause of performance penalties. Interestingly, the selection of the egress proxy operator appears to have crucial implications on performance. We also observe that client traffic over PR outperforms traffic over an unmodified connection in some cases, suggesting that the system's overlay routing can result in more optimal paths.

Overall, our study is a first step towards understanding the impact of large-scale, well-provisioned, privacy-enhancing services such as iCloud Private Relay

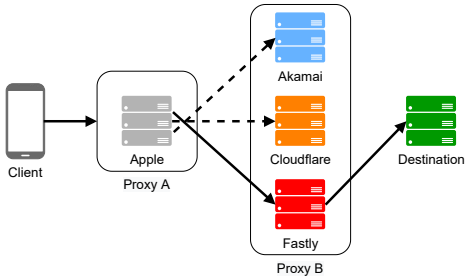


Fig. 1: Overview of the iCloud Private Relay architecture. Client traffic passes through two proxies: i) Proxy A operated by Apple; and ii) Proxy B operated by one of Akamai, Cloudflare, or Fastly.

on Internet performance. To increase the impact of our study and allow for reproducible comparisons, we release our measurements and source codes[1].

2 Background and Related Work

2.1 iCloud Private Relay Architecture

Apple launched the PR service during its Apple Worldwide Developers Conference (WWDC) in 2021 [4]. The service employs a multi-hop proxy architecture, also known as a Multi-Party Relay (MPR) [19]. The architecture provides privacy benefits by decoupling the users' network identity (i.e., the client IP address) from their Internet usage (i.e., the destination servers). This is accomplished by the client setting up two nested tunnels: the first to an ingress proxy (Proxy A in Figure 1), operated by Apple, which provides authentication and localization; the second to egress proxy (Proxy B in Figure 1) operated by one of Apple's infrastructure partners (currently including Akamai, Cloudflare, and Fastly), which in turn connects to the destination server(s) on the client's behalf. Proxy A only has visibility into the client's IP address and cannot inspect the encrypted and tunneled web traffic. Proxy B knows the servers that the clients connect to, but cannot see the client's IP address. Likewise, the destination server does not see the client IP addresses as connections are initiated by Proxy B. PR is currently limited to Apple-specific applications (i.e., Safari).

The PR architecture relies on well-known web protocols rather than custom protocols. The connection to Proxy A uses QUIC by default, with a fallback to HTTP/2 and TLS if the QUIC connection fails or is blocked. The connection to Proxy B defaults to HTTP/3 and MASQUE [18,17], which allows building efficient QUIC connections over a QUIC proxy. If HTTP/3 is not supported, the connection to Proxy B falls back to the classical HTTP CONNECT over TLS.

The PR does not act as a classical VPN and handles the traffic coming uniquely from the Safari web browser. Only HTTP(S) browser traffic goes

through the PR, while we notice that, in the case of audio/video calls, WebRTC traffic (RTP, DTLS, and STUN/TURN protocols) is not captured by the PR. The traffic of other applications in the system uses classical routing too, including other browsers and mail clients. Interestingly, the `curl` command-line facility uses PR, but only for clear-text HTTP traffic. The fact that only some applications support PR is a problem, since PR may give users a false sense of privacy while routing only a share of their traffic to the PR tunnel.

Moreover, the usage of such an architecture will impact the efficacy of existing Internet services. For instance, services that rely on client IP address information for localizing content (i.e., IP geolocation) no longer have access to clients' actual IP addresses. Other services that require insight into user traffic, such as middleboxes that provide content filtering (e.g., corporate networks or parental control services) will be unable to access user content. Lastly, the additional hops introduced by the service may hamper performance, as we investigate in this paper.

2.2 Related Work

Onion routing [24] and Tor [10] are perhaps the most similar systems compared with PR in terms of privacy goals for Internet traffic. These systems enhance users' privacy by obfuscating sender and receiver endpoints using ad-hoc "circuits" to transit user traffic. iCloud Private Relay differs in that it utilizes fewer relay hops (Tor's default is three, PR uses two), and the relays are operated on well-provisioned commercial infrastructure rather than on volunteers' systems. PR also differs from Tor in that it uses the standard HTTP and QUIC protocols rather than a custom protocol, arguably making PR more difficult to block and/or censor.

Several works have studied these privacy-enhancing services, investigating possible attacks against users' privacy. Different website fingerprinting techniques, for example, have been tested against the Tor network in [22,27,6]. In terms of performance, some studies [15,2] benchmark popular VPN services as well as Tor, finding major impairments in some scenarios. We share a similar goal with these related efforts, focusing on iCloud Private Relay.

A single work studied specifically PR [16]. The authors analyzed the network infrastructure that iCloud Private Relay has been deployed on and highlighted the geographic footprint of the service. In contrast, we study PR from the Internet performance point of view, shedding light on possible impairments users face when using the service. Indeed, our work focuses on performing an empirical evaluation of the impact on performance caused by the PR architecture, differently from [16] which focuses on the study of PR's architecture, uncovering its ingress and egress points location (geographically and network wise).

3 Testbed and Dataset

We design three measurement campaigns aiming at quantifying iCloud Private Relay performance from different perspectives. Our experiments have been per-

formed from three locations using three identical Apple MacBook Pro laptops running macOS Monterey. We deployed the laptops at three University networks, connecting them to the Internet through Gbit/s Ethernet links. Two laptops are deployed in large European cities, Lyon in France and Trieste in Italy. The third laptop is deployed in Hawaii (USA). Note that, although our study does uncover some of the potential impacts that PR has on network performance, our locations cannot be considered representative of the entire internet. Instrumenting more locations for extending our findings, e.g., by hosting probes in distributed cloud infrastructure, is left for future work.

We fully automate the experiments through custom-made testbed scripts. Common to all measurement campaigns is a script in the *AppleScript* format that automates the activation and deactivation of the Private Relay functionality of the laptop. All these scripts are contributed to the community to allow others to extend and validate our findings. We set up PR with the default option that preserves user location as much as possible.

Ethical Considerations. During our measurements, we took care to avoid harming the crawled web services. Considering that the targets of our analysis were some of the most popular websites and CDNs in Western countries, our belief is not to have caused an overload on the servers or any undesirable side effects.

3.1 Throughput Measurements

Active throughput measurements are a common tool to measure the speed of the slowest segment (the bottleneck) between a test device and a server deployed in the network. Modern speed tests commonly aim to deploy their servers in a region close to end-users, under the assumption that the bottleneck will be located at the access network. While these tests are not always representative of the user's experience [25], they can spot the performance impact caused by i) traversing additional middleboxes, i.e., iCloud PR's proxies, and ii) taking a different path between the user and the test infrastructure of the speed test service.

We perform active throughput measurements using Ookla's Speed Test service [14], one of the *de facto* standards for Internet speed test measurements. We instrument our machines to automatically perform speed tests by accessing Ookla's web page. Doing so requires i) accessing the web page; ii) detecting Ookla's privacy banner and accepting it (in Europe, not in the US); and iii) starting the test by clicking on the "GO" button. We automate this process using Selenium [20] tools and instrumenting the Safari browser. We run 200 speed tests from each location, half with PR enabled and half without PR.

3.2 Bulk Downloads

Architecturally, PR achieves privacy by decoupling information on users and the services they access. When iCloud Private Relay is enabled, all Safari traffic goes through the system by default. This has implications not only for web browsing but also for downloads of large files – i.e., performing bulk downloads of data

through HTTP. We measure PR performance on bulk downloads using the `curl` command-line tool. When Private Relay is enabled, `curl` traffic uses it, allowing us to easily test HTTP downloads in isolation. We use `curl` to download a 1 GB file several times. We select a 1 GB test file made available on the Hetzner CDN, a standard file used for evaluating content distribution speeds [11]. From each location, we download the test file 200 times with and without Private Relay and record the download time.

3.3 Web Measurements

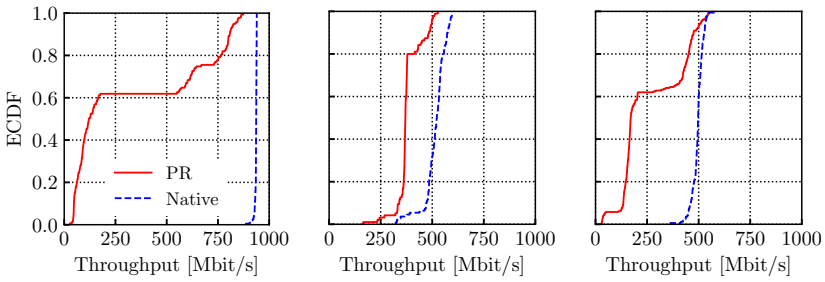
iCloud Private Relay is mainly designed to allow web browsing with stronger privacy guarantees. Our goal is ultimately to study to what extent Private Relay impacts the user's perceived performance and, in turn, its implications for web QoE. To this end, we instrument Safari to visit a set of web pages automatically and collect statistics regarding page loading. We target the 100 most popular websites in each country according to the public ranking provided by SimilarWeb analytics [21].

We use the BrowserTime toolset to automate the visits to the websites and the collection of the statistics [23]. For each website, we run five visits with and without PR enabled. Out of each visit, we collect statistics about each HTTP transaction carried out during the page loading. Essential to our analysis, we collect the Page Load Time (also called `onLoad` time) that we use as a practical proxy for measuring the web performance. Page Load Time represents the time elapsed between the beginning of the visit and the instant when the last object of the web page is retrieved. The Page Load Time has previously been shown to be correlated with users' QoE [9].

Finally, note that in our experimental campaign, we do not measure explicitly the end-to-end RTT. Indeed, our measurement infrastructure cannot observe the layer-4 RTT, as we rely on browser instrumentation. Measuring the RTT poses some challenges in the case of tunneled traffic (such as PR), e.g, one could instrument the SO kernel to monitor TCP statistics. This is by no means trivial, in particular considering the proprietary software offering PR. We thus focus on user-perceived quality, showing higher-level metrics such as Page Load Time or Throughput, leaving these additional aspects for future work.

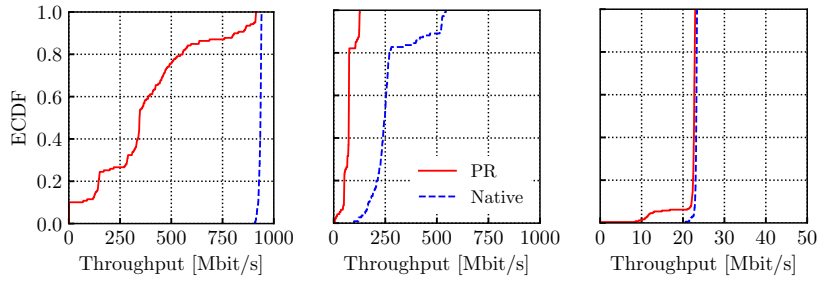
4 Results

We now present results across the three workloads. We observe that, in general, PR negatively impacts performance, particularly for scenarios that require long-lasting network flows, i.e., bulk download and speed test measurements. Further, in these experiments, PR usage results in a higher level of variability in performance, even for stable and fast Ethernet network connections. Interestingly, these takeaways do not apply across all results: in one case, i.e., bulk download in France, we observe that PR outperforms an unmodified connection.



(a) France (b) Italy (c) US

Fig. 2: Download throughput measured with speed test measurements.



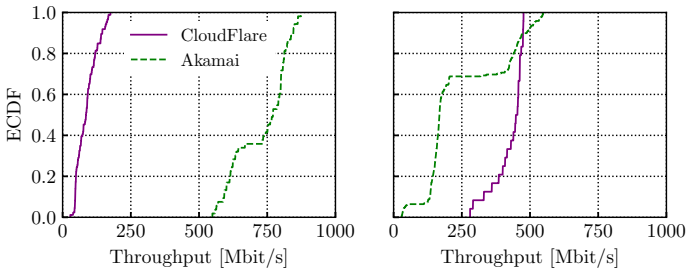
(a) France (b) Italy (c) US

Fig. 3: Upload throughput measured with speed test measurements. Note the different x-scale for the US.

4.1 Throughput

We first evaluate the performance impact of PR on active throughput measurements using Ookla's infrastructure. Overall, we expect performance to be, at the very least, impacted by the overhead of the PR tunnels, as disclaimed on Apple's support website [3]. Here, we look to quantify this overhead, shedding light on the incurred performance penalties.

Figures 2 and 3 show the Empirical Cumulative Distribution Function (ECDF) for the downstream and upstream throughput with and without PR, respectively. Overall, we observe that performance is drastically impacted across the vast majority of scenarios, both for downstream and upstream throughput. This impact is particularly significant for our France measurement location (Figures 2a and 3a), where measurements experience a median speed reduction of 87% and 63% respectively. The sole scenario that does not present an evident reduction in performance is the uplink throughput in our US measurement location, where performance is capped at around 23 Mbit/s for both experiments. This suggests that the bottleneck, in this case, is most likely to be found in the



(a) France (b) US

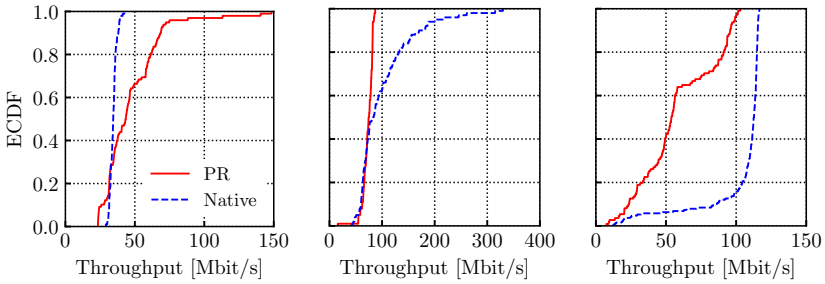
Fig. 4: Download throughput with PR and different Proxy B operators.

path between the client and Proxy A. In sum, PR seemingly does not impact performance when the client-side connections are the bottleneck.

Interestingly, we observe that speed tests performed when PR is enabled result in a much higher performance variability. For example, we observe that in multiple configurations, in particular for downstream experiments in France and the US (Figure 2a and 2c, respectively), experiments result in bimodal speed distributions, possibly caused by either ephemerally congested paths or congested proxies that negatively impact performance in a subset of experiments.

We investigate this aspect further in Figure 4, where we dissect throughput distribution according to Proxy B selected as the egress node by PR. Sattler et. al [16] found that Proxy B selection changes multiple times in a day. For France, we observe that all speed tests achieving throughput below 200 Mbit/s are those using a Cloudflare-owned Proxy B, while the faster ones are all using Akamai's Proxy B. In the US, we observe the opposite scenario, with CloudFlare Proxy B leading to better performance compared with Akamai, even if the two distributions partially overlap. We do not report the figure for Italy as all experiments for this case resulted in an Akamai Proxy B egress, leading to the performance shown in Figure 2. We also observe that when PR is in place, the Ookla's measurement server is often further from the user than without PR for both Italy and France. With native connection, the speed test is served from a server within 120 km, while, with PR, the server is $200-300$ km far away. We detail this in the Appendix. In a nutshell, the choice of egress node has paramount implications on the achieved throughput, and this choice is not under the user's control.

Overall, these results appear to confirm Apple's disclaimer that PR can negatively impact speed test performance. Apple justifies this performance loss to the normal behavior of speed test experiments. In particular, they state that "Private Relay uses a single, secure connection to maintain privacy and performance. This design may impact how throughput is reflected in network speed tests that typically open several simultaneous connections to deliver the highest possible result." To verify whether the performance loss experienced can be solely linked to the use of multiple connections, in the next section, we replicate



(a) France (b) Italy (c) US

Fig. 5: Average download speed during downloads of a large file.

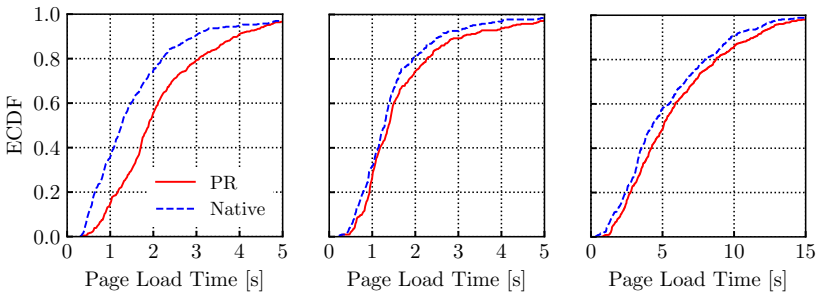
the performance comparison in a scenario where only a single connection is used, i.e., the single file bulk download over HTTP.

4.2 Bulk Download

We now compare the performance achieved with and without PR downloading a large 1 GB file from a CDN. Figure 5 shows the obtained results for our three locations. We observe that, although to a different extent, the US and Italy locations still experience a similar negative performance impact when downloading the file via PR, even when using a single network flow.

For the US case, we see that PR reduces performance by 53% in the median case. For the Italian case, we notice that median throughput is usually similar with and without PR, except for the tail of the distributions. In particular, in this case, PR performance is very stable, never exceeding around 80 Mbit/s. In contrast, similar tests without PR can exceed 300 Mbit/s. We conjecture that, for the Italian location, the traffic traversing PR takes a path where traffic cannot exceed 80 Mbit/s. This can be due to congestion or route peering arrangements, even if we cannot precisely pinpoint the root causes for these differences.

Most interestingly, we observe that for our experiments in France, results show very similar behavior, but with inverse outcomes, i.e., PR downloads experience *higher* throughput while non-PR traffic is capped at around 35 Mbit/s. We investigate this behavior further using `traceroute` and we observe that the selected CDN node changes when connected to PR and, consequently, packets follow different routes. More precisely, we observe that, when PR is not enabled, packets traverse an operator (GEANT) that is otherwise not observed when using PR. This suggests that, when not using PR, packets encounter a bottlenecked link, which is at the root of the impaired performance. Effectively, the presence of PR-induced overlay routing overrides default routes taken by client traffic in the French location. These results call for further investigation of the routing of traffic when PR is enabled, which we leave for future work.



(a) France (b) Italy (c) US

Fig. 6: Page Load Time distribution.

4.3 Web Browsing

Lastly, we study the performance impact that the use of PR has on web browsing. To this end, we measure the Page Load Time for the top-100 ranked websites in each of the three test locations. According to Apple's claim, web experience should remain "fast" even with PR active, while our results suggest that performance varies. Figure 6 shows the distribution of Page Load Time for the three locations, with and without PR active.

At a first glance, we observe that PR consistently introduces an additional delay to page loading regardless of the location. The performance impact is particularly evident in France, where the median Page Load Time increases by almost 60%. For the other locations, we observe a more moderate increase of 7% and 17% for Italy and the US, respectively. Interestingly, our measurements show that TLS handshake time increases by 9-14% when using PR, depending on the location. This increase likely impacts the page-loading process negatively, but only partially explains the performance degradation with PR.

Overall, while web performance is not as heavily impacted as throughput measurements, the claim that the architecture of PR exclusively impacts speed tests might be reductive. More detailed experiments might be required to shed light on the root causes of the additional page load times. We speculate that these results are caused by the additional overhead caused by the traversal of multiple middleboxes and the necessarily longer path packets must travel. Network latency is known to impact web QoE directly [8,12], and even a small deterioration in page load time has a large impact on the web ecosystem [7].

5 Discussion and Open Questions

We now elaborate on the limitations of our findings and possibilities for future work. In general, answering the questions listed below requires further measurements from different locations. While our measurements include several campaigns, they cannot be considered representative of the whole Internet. Indeed,

we will extend our measurements in future work by deploying new probes i) in other locations – e.g., by leveraging cloud infrastructure in more countries, and ii) in heterogeneous scenarios – e.g., by deploying probes in wireless and cellular networks. Ultimately, we contribute our scripts to the community to allow others to validate and extend our results in autonomy, considering novel scenarios.

5.1 Overriding Routing

In Section 4.2, our results illustrate how the use of PR can lead to better performance compared to native connectivity in particular cases. We find that overlay routing appears to result in the avoidance of a congested network when using PR in one particular case. This result demonstrates one of the potential impacts that Multi-Party Relay architectures can have on network performance: The chosen traffic paths are dictated by the combination of the user's ISP, Apple, and Apple's infrastructure partners for Proxies B, rather than simply the user's ISP and the destination server. In effect, at the ISP, packet routing mechanisms with PR will significantly diverge from existing patterns. Where, today, ISPs observe a fan-out pattern, routing client traffic to the many destination networks on the Internet, networks with high usage of PR in the future will see a many-to-one pattern, with all client traffic routing to a single destination network (Apple or whoever is operating the ingress proxy of a multi-party relay).

Indeed, our initial experience while running measurements in mobile networks in France and Italy shows that performance metrics (with and without PR) are more susceptible to variations throughout the day. Additional factors, such as the cellular network load during the day, make the study of PR on mobile networks largely more complex. Longitudinal measurements in such environments are needed to draw robust conclusions and we will pursue that in future work.

5.2 Localization

iCloud Private Relay is designed to prevent destination servers from observing client IP addresses. Clearly, this design negatively impacts the ability of IP geolocation services to map clients to their geographical location. These services are widely used by content providers to localize users and determine access rules based on geographical constraints. The PR architecture aims to minimize this issue by roughly localizing the client using Proxy A, and carefully selecting the Proxy B egress based on the location that the client is purported to be. This would preserve, at least roughly, the geographic location of the user from the server's point of view. To support IP geolocation services in mapping the users' geographical location, Apple publishes Proxy B IP addresses along with the location of the users aggregated through them [5].

In many cases, low-fidelity location information is sufficient to provide localized content. Unfortunately, some services require very accurate location information to serve content (e.g., live streaming of sporting events), which may not be possible using services such as PR. Further study is required to study the tradeoff between privacy and usability in terms of localization. Additionally,

previous work [16] has shown that the IP-to-location mappings offered by Apple's partners are not always a direct representation of the physical location of the proxy holding the given IP. This is done to overcome the lack of PR proxies in certain regions of the world. This could impact performance for users who connect to the PR infrastructure from locations that are not physically served by it. The network paths would be extended beyond their geographical location, adding latency to communications and crossing national borders.

5.3 Cost

While the PR design seems beneficial for privacy, the real benefits have been left unquantified and largely unexplored. Future work is necessary to understand the benefits offered by such as system. This is particularly true considering the inherent tradeoffs that Multi-Party Relay architectures have on network traffic and the capability required to process it: In PR, clients' traffic passes through multiple middleboxes in order to achieve the privacy guarantees associated with decoupling network identity from behavior. This has implications on performance, at the center of this paper, as well as on energy consumption (e.g., due to the additional servers and the multiple layers of encryption they have to handle). For example, by nesting encrypted channels as the PR architecture does, Proxy A could be wasting significant computing resources "double encrypting" traffic. To avoid this overhead, QUIC-Aware Proxying Using HTTP has been proposed, where Proxy A simply moves the traffic along the path towards Proxy B without double encryption [18,17]. Other similar optimizations are likely to be introduced as the architecture becomes more mature and more widely adopted.

6 Conclusions

Apple's iCloud Private Relay is one of the recent attempts at deploying Multi-Party Relay architectures at scale. Given Apple devices' pervasiveness and the company's push towards privacy, it is possible that this architecture will be quickly adopted as the *de facto* standard for privacy-oriented network architectures. In this work, we present a first study of the impacts that PR architecture can have on users' performance. Through experiments across three locations in France, Italy, and the US, we find that PR not only impacts active throughput measurements but also negatively affects page load time and file download, indicating potential impacts on the users' web QoE. We show for example that PR substantially changes the paths taken by traffic (e.g., during speed tests), impacting performance. Our paper sheds light on new problems and calls for further research on how to avoid them when deploying privacy-preserving services.

This work opens up a number of potential future venues to explore Multi-Party Relay architectures such iCloud Private Relay, not solely in terms of performance, but also across multiple dimensions such as privacy-costs tradeoffs, content access, and the impact on network routing at large. To engage the community to search for the answer to these questions, we release the source code of the software used to perform the experiments presented in this paper.

Appendix

In this Appendix, we break down the distance between the user and Ookla's speed test measurement servers, with and without PR. In the following two tables, we show the measurement server chosen by Ookla, detailing its location and distance from the testing location. We report data for the Italian and French locations and separate the cases with and without PR. We omit the US location as, in all cases, the measurement server is located at the same location, i.e., Hawaii.

When PR is in place, it is more likely that the speed test server is far away from the client. For example, for the Italian location, without PR, speed tests are served within 120 km, while with PR, servers are at 200 km or more from the client.

Ookla obviously cannot identify the true location of the users, since its servers observe only egress IP addresses. Indeed, hiding the users' IP addresses is the ultimate goal of PR and, as such, these differences are expected. We here show that the servers selected by Ookla when PR is enabled deliver poorer throughput figures, and our conjecture is that the root causes for such performance penalties are in the path from clients to the selected servers.

The same situation may occur with other services relying on IP geolocation, such as content providers and CDNs. Our measurements, while preliminary, show that the introduction of the PR tunnels impact performance (see our discussion on future work in Section 5).

Table 1: Share of Speed Tests served from servers in different locations. The distance from the client is reported in brackets.

	Ljubljana (70 km)	Venice (120 km)	Conegliano (120 km)	Milan (200 km)	Rome (400 km)
Native	12.7%	81.8%	5.5%	0.0%	0.0%
PR	0.0%	0.0%	0.0%	98.9%	1.1%

(a) Italy

	Lyon (0 km)	Marseille (270 km)	Nice (300 km)
Native	100.0%	0.0%	0.0%
PR	0.0%	85.2%	14.8%

(b) France

References

1. https://github.com/marty90/icloud-private-relay-experiments.
2. Mashael Alsabah and Ian Goldberg. Performance and security improvements for tor: A survey. *ACM Comput. Surv.*, 49(2), sep 2016.
3. Apple. About iCloud Private Relay. https://support.apple.com/en-us/HT212614, December 2021.
4. Apple. iCloud Private Relay Overview. https://www.apple.com/privacy/docs/iCloud_Private_Relay_Overview_Dec2021.PDF, December 2021.
5. Apple. Prepare Your Network or Web Server for iCloud Private Relay. https://developer.apple.com/support/prepare-your-network-for-icloud-private-relay/, December 2021.
6. D. Arp, F. Yamaguchi, and K. Rieck. Torben: A Practical Side-Channel Attack for Deanonymizing Tor Communication. Proc. of the ASIA CCS, pages 597–602, 2015.
7. Fast Company. How one second could cost amazon $1.6 billion in sales. http://www.fastcompany.com/1825005/how-one-second-could-cost-amazon-16-billion-sales, December 2021.
8. Heng Cui and Ernst Biersack. On the relationship between qos and qoe for web sessions. *EURECOM, Sophia Antipolis, France, Tech. Rep. RR-12-263*, 2012.
9. Diego Neves da Hora, Alemnew Sheferaw Asrese, Vassilis Christophides, Renata Teixeira, and Dario Rossi. Narrowing the gap between qos metrics and web qoe using above-the-fold metrics. In *International Conference on Passive and Active Network Measurement*, pages 31–43. Springer, 2018.
10. R Dingledine, N Mathewson, and P Syverson. Tor: the second-generation onion router', usenix security symposium, 2004.
11. Hetzner. Test-files. http://speed.hetzner.de, December 2021.
12. Anna Maria Mandalari, Andra Lutu, Ana Custura, Ali Safari Khatouni, Özgü Alay, Marcelo Bagnulo, Vaibhav Bajpai, Anna Brunstrom, Jörg Ott, Martino Trevisan, et al. Measuring roaming in europe: Infrastructure and implications on users' qoe. *IEEE Transactions on Mobile Computing*, 21(10):3687–3699, 2021.
13. Akshaya Mani, T. Wilson-Brown, Rob Jansen, Aaron Johnson, and Micah Sherr. Understanding tor usage with privacy-preserving measurement. In *Proceedings of the Internet Measurement Conference 2018*, IMC '18, page 175–187, New York, NY, USA, 2018. Association for Computing Machinery.
14. Ookla. Speedtest. http://speedtest.net, December 2021.
15. Maximilian Pudelko, Paul Emmerich, Sebastian Gallenmüller, and Georg Carle. Performance analysis of vpn gateways. In *2020 IFIP Networking Conference (Networking)*, pages 325–333, 2020.
16. Patrick Sattler, Juliane Aulbach, Johannes Zirngibl, and Georg Carle. Towards a tectonic traffic shift? Investigating Apple's new relay network. In *Proceedings of the 22nd ACM Internet Measurement Conference*, IMC '22, 2022.
17. David Schinazi. Proxying UDP in HTTP. RFC 9298, August 2022.
18. David Schinazi and Lucas Pardue. HTTP Datagrams and the Capsule Protocol. RFC 9297, August 2022.
19. Paul Schmitt, Jana Iyengar, Christopher Wood, and Barath Raghavan. The decoupling principle: A practical privacy framework. In *ACM SIGCOMM Workshop on Hot Topics in Networking (HotNets)*, November 2022.
20. Selenium. Selenium automates browsers. that's it! https://www.selenium.dev, December 2021.

21. SimilarWeb. Effortlessly analyze your competitive landscape. `https://www.similarweb.com/`, December 2021.
22. P. Sirinam, M. Imani, M. Juarez, and M. Wright. Deep Fingerprinting: Undermining Website Fingerprinting Defenses with Deep Learning. Proc. of the CCS, pages 1928–1943, 2018.
23. sitespeed.io. Documentation v16. `https://www.sitespeed.io/documentation/browsertime/`, December 2021.
24. Paul F Syverson, David M Goldschlag, and Michael G Reed. Anonymous connections and onion routing. In *Proceedings. 1997 IEEE Symposium on Security and Privacy (Cat. No. 97CB36097)*, pages 44–54. IEEE, 1997.
25. Martino Trevisan, Idilio Drago, and Marco Mellia. Impact of access speed on adaptive video streaming quality: A passive perspective. In *Proceedings of the 2016 Workshop on QoE-Based Analysis and Management of Data Communication Networks*, Internet-QoE '16, page 7–12, New York, NY, USA, 2016.
26. Martino Trevisan, Francesca Soro, Marco Mellia, Idilio Drago, and Ricardo Morla. Does domain name encryption increase users' privacy? *SIGCOMM Comput. Commun. Rev.*, 50(3):16–22, jul 2020.
27. T. Wang, X. Cai, R. Nithyanand, R. Johnson, and I. Goldberg. Effective Attacks and Provable Defenses for Website Fingerprinting. Proc. of the USENIX Security, pages 143–157, 2014.