# Path Transparency Measurements from the Mobile Edge with PATHspider

Iain R. Learmonth*, Andra Lutu†, Gorry Fairhurst*, David Ros†, Özgü Alay†
*University of Aberdeen
†Simula Research Laboratory

***Abstract*—Network operators and equipment vendors can hesitate to deploy network protocol innovations in fear of breaking connectivity for end users. To assess the potential for evolution of the protocol stack, it is important to know the existing network impairments and opportunities to work around the impairments. While classical network measurement tools often focus on absolute performance values, PATHspider is an extensible framework for performing and analyzing A/B testing between two different protocols or different protocol extensions. It thus enables controlled experiments in search of protocol-dependent connectivity problems, and to identify differential treatment. This paper presents how PATHspider can be instrumented to assess path transparency over commercial mobile networks, using the MONROE platform. We provide here proof-of-concept results from measurements in a UK commercial mobile network, and lay out our future measurement plans for PATHspider using the MONROE testbed in Europe.**

## I. Introduction

Economic incentives, the need to remain competitive, security requirements and the necessity for network address translation, have caused an increasing accumulation of middleboxes in modern communication networks. Middleboxes are especially prevalent in mobile networks, where they involve multiple layers of Network Address Translation (NAT), apply complex firewall policies, insert performance enhancing proxies [1] in a transport path, enforce censorship, and support an assortment of methods for mobile network operations. However, middleboxes often also make assumptions about the traffic and protocols used, which can lead to ossification of the protocol stack [2]. This raises questions about whether any new protocol header would be passed though a network path. It is thus imperative to understand the interaction of new protocol mechanisms with the middleboxes active along a path. The increase in pervasive transport encryption, partly as a reaction to middleboxes (e.g., QUIC [3]), also increases the complexity of this space, because, while this can help avoid application ossification, it can also impact operational support [4].

The Measurement and Architecture for a Middleboxed Internet (MAMI) European project seeks to explore how the network can be enabled to support better evolution[1]. As a starting point, this project focused on developing tools for A/B testing of path transparency in the current Internet, in the form of PATHspider [5]. This is functional testing as opposed to bandwidth and performance measurement and provides the basis for designing new methods to enable useful coexistence between encrypted protocols and middlebox functions. PATHspider has been used to probe from multiple cloud vantage points to web servers [6] [7] and by peer-to-peer clients [8] to examine failures negotiating Explicit Congestion Notification (ECN) [9]. A flexible plugin design enables PATHspider users to design and incorporate their own custom tests into the tool. Recent updates have added plugins for TCP Fast Open (TFO) [10] and Differentiated Services Code Point (DSCP) [11].

While previous studies using PATHspider have focused on the core of the Internet, in this paper we leverage early access to the MONROE platform [12], [13] and customize PATHspider to enable measurements across mobile broadband paths. MONROE is the first open access hardware-based platform for independent, multihomed, large-scale experimentation in Mobile Broadband (MBB) heterogeneous environments. The testbed comprises a large set of custom hardware devices, both mobile (e.g., via hardware operating aboard public transport vehicles) and stationary (e.g., volunteers hosting the equipment in their homes), each with three multihomed interfaces to different MBB operators using commercial grade subscriptions.

The rest of the paper is organized as follows. Section II reviews other measurement tools and discusses their suitability to run in a mobile broadband context. We then provide an overview of the PATHspider architecture in Section III. Section IV describes the MONROE platform and how we customized PATHspider to work in this environment. We then detail its functionality and use in MONROE in Section V. Furthermore, we exemplify the wide range of experiments PATHspider on MONROE enables us to perform (Section VI) and showcase initial results (Section VII). Section IX concludes the paper.

## II. Related Work

Existing active measurement platforms, such as RIPE Atlas [14], OONI [15], or Netalyzr [16], were built to measure performance and connectivity between a pair of endpoints under specific conditions. The results from measurements can be compared to simulate A/B testing. However, path characteristics can change, and results accuracy decreases when different tools, or even the same tools, are used at different times to perform the measurements.

OONI produced mobile measurement applications for Android and iOS devices and Netalyzr have produced a mobile

[1]https://mami-project.eu

application for Android devices, however these applications are limited because they execute in a sandboxed environment. Detailed network measurements typically require application access to raw sockets, prohibited in both Android and iOS (these restrictions can be worked around by "rooting" or "jailbreaking" the device [17]). TraceboxAndroid [18] is a third mobile application based on Tracebox [19] that was developed to measure path transparency. Tracebox [19] performs traceroute-like probes, incrementing the TTL for each packet sent and then analyzing the ICMP quotations. This methodology also requires raw sockets, and the TraceboxAndroid authors noted that this was only a proof-of-concept and they did not expect this to become a scalable measurement solution.

Given the above, there remains a pressing need for new measurement data and tools that can inform development and deployment of new protocols.

## III. PATHspider Architecture

PATHspider [5] comprises four components, which we illustrate in Figure 1: workers, a configurator, an observer and the merger. These components run on a vantage point, a host connected to the Internet with a specific routing origin. By using multiple vantage points, it is possible to gain insight into more networks as more are traversed. It can also become possible to reduce levels of noise in the data collected by combining datasets from multiple vantage points.

The workers generate test traffic towards the target host from a vantage point, which is passively observed by the observer component to acquire measurements. Measurements are defined by a combination of vantage point, target host and configuration sequence number.

The operation of the A/B test depends on the PATHspider plugin. The behavior of workers may either be modified by the configuration sequence number or can remain constant while the behavior of the native network stack is modified by the configurator. In the first case, no synchronization is required and measurements may be made continuously while the jobs are processed as they are fed. In the second case, connections must be synchronized to ensure that the traffic is generated using the required network stack state for each measurement.

The merger combines results from the traffic generator, including any application layer information (e.g., HTTP response code), with the results from the observer including network and transport layer information. These records are then combined again for each job to understand whether the collected measurements achieve connectivity after using a specific feature in an A/B test (e.g., after attempting ECN negotiation or use of a specific DSCP or the TFO option).

PATHspider provides an extensible framework for measurements. Each measurement is customized through plugins. This framework makes it easy to deploy measurement campaigns for the evaluation of path transparency for new protocols and protocol extensions through plugins. For a more detailed documentation on PATHspider we refer the reader to the online resources[2]. At the time of writing, we are in the process of publishing the PATHspider results to the Path Transparency Observatory (PTO) [20] operated by the MAMI project.

[2]https://pathspider.net/

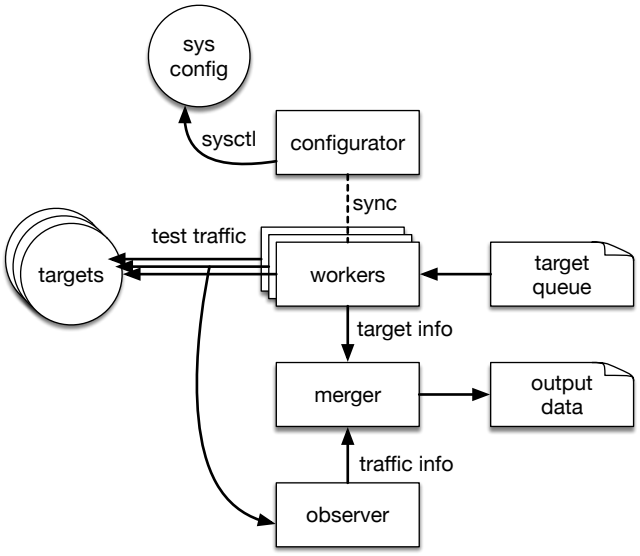


Fig. 1. Block diagram illustrating control flow and flow of data between PATHspider components.

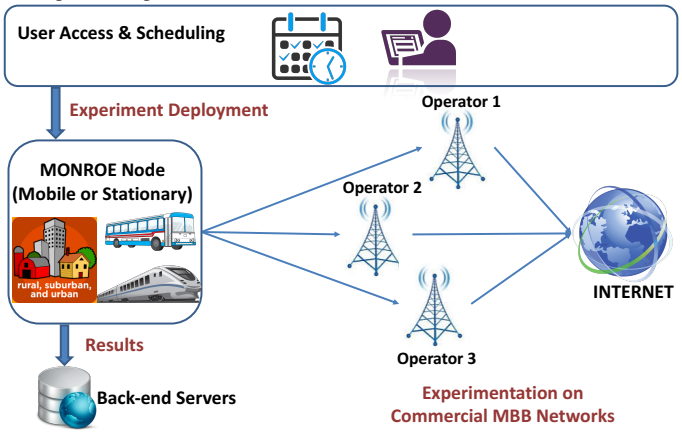


Fig. 2. Key components of the MONROE platform. MONROE users access resources and deploy their experiments via the User Access and Scheduling function. Measurement results are periodically synchronized to external repositories in the back-end.

## IV. The MONROE Platform

The MONROE platform [13] is the first open access hardware-based platform for independent, multihomed, large-scale experimentation in commercial MBB heterogeneous environments. Figure 2 provides an overview of the main building blocks. All software components used in the platform are open source and available online[3].

The platform comprises 250 measurement hardware devices called MONROE Nodes[4]. Each MONROE node is a Linux-based programmable device (the PCEngines APU board[5])

[3]https://github.com/monroe-project/

[4]At the time of writing, 200 nodes were deployed, with deployment scheduled to completed by June 2017.

[5]APU: https://www.pcengines.ch/apu2c4.htm

that uses three different 3G/4G modems[6] (LTE CAT6) to multihome to three MBB operators, using one modem for each carrier. The nodes are deployed in heterogeneous environments, including mobile (e.g., nodes deployed on public transport vehicles) and stationary ones (e.g., volunteers hosting nodes in their homes), as we show in Figure 2.

At the time of writing, MONROE operates measurement nodes in five European countries (Spain, Italy, Sweden, Norway and the UK). Nodes are multihomed to (up to) three different MBB operators using commercial subscriptions in each country. In other words, each MONROE node is equivalent to three measurement vantage points (one for each MBB network) that can be used simultaneously for experimentation. Each node connection aims to mimic an end-user connected to three operators using the same hardware, thus providing a controlled and configurable platform for benchmarking the mobile service. Each node has a volume quota available with each subscription, which is aligned with current commercial offers from the operator (e.g., in Sweden the maximum available data quota is 200GB/month, while in Spain we have a maximum limit of 10GB/month). The MONROE platform therefore monitors and regulates the volume of data that each experiment is allowed to exchange by a node.

The node software is based on Debian GNU/Linux *"stretch"*[7]. Using Linux provides accessibility of the source code, flexibility and community maintenance to ensure interoperability with other systems and flexibility in the hardware required to support research and implementation of protocols.

Each Node runs: (i) *the management software* that ensures the node remains operational (e.g., MBB modems correctly configured and connected, routing enabled) and enables remote updates of all the other software components, (ii) *the maintenance software* that monitors operational status and reduces the need for manual maintenance and (iii) *the experimentation enablers*, facilitating experiment deployment (via the scheduler client) and feeding rich context information to the experiments.

The management software provides: (i) a Device Listener to detect, configure and connect USB network interfaces, (ii) a routing daemon that uses DHCP to acquire an IP address and set up routing tables and (iii) a network monitor to monitor interface state, check connectivity and configure default routes. The **node maintenance software** integrates components that monitor the node status and trigger actions to repair or reinstall when malfunctioning. A *system-wide watchdog* ensures that all core components (node management) are running. The **experimentation enablers** include the scheduling client and the services for external experiments. Experiments running on the platform use the Docker[8] light-weight virtualized environment to provide containment of user experiments. Prior to their scheduled run time, the scheduler deploys the containers to the nodes the user previously selected. This allows us to controls the access of external users to a node. The *metadata broadcasting service* runs continuously in the background and relays metadata through ZeroMQ[9] in JSON format to experiment containers running on the node. The experiment running inside the container can then subscribe to different metadata topics from the ZeroMQ socket and coordinate running measurements on the available mobile connections on the node.

User Access to the MONROE nodes is through a web portal, which allows an authenticated user to use the MONROE scheduler to schedule and deploy experiments running within Docker containers which are hosted at the MONROE Docker repository. This enables the MONROE user to check for available nodes in the platform and get exclusive access to the number of nodes requested (i.e., no two experiments run on the same node at the same time).

The results from each experiment are periodically transferred from the nodes to a repository at a backend server (see Figure 2). The user can retrieve the experiment results from the User Interface, where the MONROE system provides a link to download the results of each measurement performed on each node from the backend server.

## V. PATHspider on MONROE

On cloud platforms, the PATHspider authors have used Vagrant[10] and Ansible[11] for orchestration of vantage points. On the MONROE platform, each node can provide multiple vantage points by being connected to multiple mobile broadband providers. To enable PATHspider to execute on MONROE we created a Docker image, as is required by the platform, containing PATHspider and its dependencies[12]. In order to run on deployed nodes in the MONROE platform, each container has to pass the certification phase, after which it becomes available from the MONROE public repository for all MONROE users to access and deploy.

The MONROE User Access and Scheduler interface enables users to pass JSON serialized options to the container at runtime, which gives flexibility when defining the configuration of the experiment. In our case, this allows the Docker image to contain all the available plugins and for those plugins to be customized for an experiment. To ease submission of PATHspider experiments to the MONROE platform and to provide a framework for quickly downloading results, we built a command line interface for the scheduler[13] .

While PATHspider was developed to be a generalized solution to the problem of measuring path transparency, running PATHspider on the MONROE platform presented issues we have not previously considered and that we needed resolve. PATHspider uses the native network stack where possible to produce results. One hurdle to running PATHspider was that

[6]MC7455 miniPCI express (USB 3.0) modem: https://www.sierrawireless.com/products-and-solutions/embedded-solutions/products/mc7455/

[7]https://www.debian.org/releases/stretch/

[8]https://www.docker.com/

[9]http://zeromq.org/

[10]https://www.vagrantup.com/

[11]https://www.ansible.com

[12]The sources are at https://github.com/mami-project/pathspider-monroe

[13]The sources for the MONROE command line interface are available at https://github.com/ana-cc/monroe-cli

MONROE Nodes are typically multihomed, so it is impossible to know the number and name of interfaces at a node ahead of deployment. To tackle this issue, a wrapper script was integrated in the Docker image that automatically detects and enumerates the available interfaces. Then, the requested measurement runs on each interface in sequence. PATHspider then configures test traffic generation and packet capture according to the local routing table. PATHspider connection helpers provide the source addresses required to bind to each interface. We then use another wrapper script in the Docker image to write the metadata records to every output file with interface metadata for later use in analysis. This allows us to determine important context information, including the first globally routable IP address for a given interface that operates behind a NAT router, the operator to which the interface connects and the country within which it is located.

PATHspider has traditionally operated on cloud platforms with reliable network connections. On mobile networks, especially with mobile nodes that change location, it is possible that connections become unstable or fail. Since PATHspider tests for functionality across the path, not performance, it is important to eliminate results that arise from interface status changes. An event listener therefore checks each available interface and triggers if an interface goes down, allowing the results impacted by these changes to be invalidated.

Following initial proof-of-concept measurements (Section VII), we discovered several other issues we aim to tackle when customizing PATHspider for MONROE. We detail these considerations and how we plan to address them in Section VIII.

## VI. PATHspider Experiments

PATHspider experiments focus on measurements that can help assess the feasibility and/or the deployment of new network protocol techniques and further inform their design. Within the MAMI project, we are particularly interested in using the PATHspider plugins to detect path impairments, with a focus on MBB networks.

Using PATHspider in MONROE, users can perform measurements for a selection of Internet protocol mechanisms and test several known issues, as follows:

- ECN [9]
- DSCP [11]
- TFO [10]
- Support for new UDP-based protocols (e.g., QUIC [3])
- HTTP/2 [21] and TLS Extensions [22]

For example, across a mobile network, experimenters can measure ECN connectivity failure when the use of ECN is attempted. More than this, PATHspider can identify whether ECN is effective and can evaluate the ability to reduce network delay that can arise due to bufferbloat. These type of measurements can also detect connectivity failure when the initial negotiation succeeds, but subsequent required signaling fails.

At the transport layer, PATHspider can evaluate the support for new end-to-end TCP mechanisms. For example, using the TFO plugin, we can identify where TFO can be successfully negotiated to a web server by the exchange of a TFO cookie and subsequently where the use of TFO is successful (i.e., where data sent on the first packet is acknowledged). Extending PATHspider to support UDP may allow it to explore the potential for deploying methods such as QUIC [3] or PLUS [23], and whether middleboxes in the network disrupt a UDP-based transport. We leave this for future work.

At the application layer, PATHspider enables the exploration of HTTP/2 and TLS extensions or the use of the Application Layer Protocol Negotiation (ALPN) and Next Protocol Negotiation (NPN) extensions to negotiate use of HTTP/2 when connecting to web servers. These measurements can detect connectivity failures due to the use of the extensions and whether or not it is possible to negotiate HTTP/2 with the server across a mobile network. A study using PlanetLab vantage pointspreviously obtained similar measurements [24].

TABLE I
Summary Statistics for Initial Measurements of ECN Path Transparency with a Mobile Broadband Access Network to 6264 hosts)

| | EE (AS12576) | | Hetzner (AS24940) | |
|---|---|---|---|---|
| **Description** | **hosts** | **pct** | **hosts** | **pct** |
| Offline hosts | 24 | 0.38% | 67 | 1.07% |
| Connected without ECN; | **6228** | **99.43%** | **6179** | **98.64%** |
| ... also connected with ECN | 6209 | 99.69% | 6175 | 99.93% |
| ... and negotiated ECN | 5900 | 94.73% | 4334 | 70.14% |
| ... failed to connect with ECN | 19 | 0.31% | 4 | 0.06% |
| Transient failure † | 12 | 0.19% | 18 | 0.29% |

† Transient failures are cases where the connection was successful in the experimental case, with ECN enabled, but not in the baseline case, without ECN. We report these numbers to give an indication of the noise present in the results due to congestion or other transient issues.

## VII. Proof-of-Concept Experimental Results

Using the customized version of PATHspider for MONROE, we instrument the MONROE nodes operating in the UK to run a proof-of-concept measurement campaign. This initial measurement campaign was used to determine the factors that may need to be considered specifically for the measurement across MBB access networks (which we discussed in Section V). For the purpose of this study, we investigated ECN-dependent connectivity failure for the EE[14] 4G network in Aberdeen, Scotland (AS12576). We compared these results to results from a Hetzner[15] datacenter in Germany (AS24940, not a MONROE node). We collected our results for both experiments on the 31st March 2017, within hours of each other.

We tested the connections from each vantage point to a total of 6,264 unique targets (resolved from the top 4,500 domains in the Alexa top 1 million[16] on 31st March 2017). We only investigated IPv4 targets because the EE operator platform did not provide IPv6 connectivity. Where operators support IPv6,

[14]http://ee.co.uk/
[15]https://hetzner.de/
[16]http://www.alexa.com/topsites

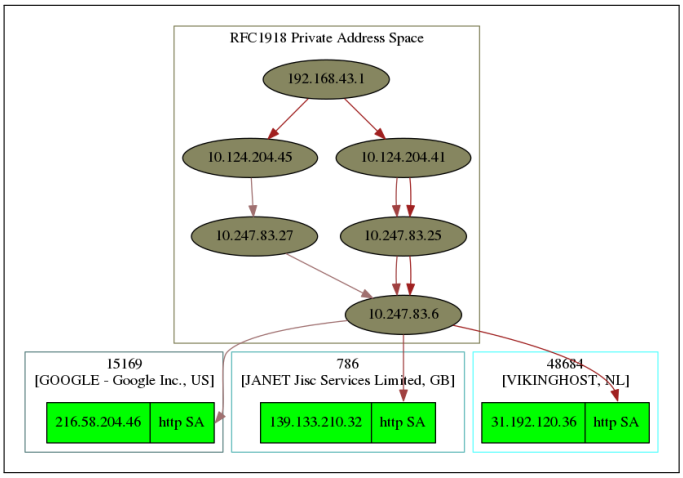


Fig. 3. Visualization of a traceroute performed to three websites (left-to-right: `google.com`, `erg.abdn.ac.uk` and `pornhub.com`) using TCP SYN packets for port 80 from the EE network.

collecting IPv6 results will be as simple as adding IPv6 target addresses to PATHspider's input list.

The baseline test requested the root resource for the domain using HTTP on TCP port 80 for each target. Immediately following this, a second request was made using a new TCP connection to the same target for the same resource, this time requesting ECN negotiation. Table I shows a summary of the results from these measurements.

Our results show no evidence of ECN-dependent connectivity failure in the EE network. There is, however, a discrepancy between the number of targets negotiating ECN in each network. The results from the Hetzner vantage point are consistent with previous studies with PATHspider [7]. Upon investigation against further targets with known configurations that we control, we discovered that the EE network may employ a TCP terminating proxy (e.g., a web proxy or performance enhancement proxy) that negotiates ECN regardless of support in the target.

This does not explain that only 94.73% of the targets that connected when attempting to negotiate ECN completed the negotiation, leaving 5.27% of hosts that did not successfully negotiate ECN though the connection succeeded. Upon further investigation, we determined that these were primarily targets associated with websites hosting pornographic content. In the UK, ISPs have received pressure from the Government to block such sites by default [25]. EE complies with this policy, and any attempts to access these sites were redirected to a block page. This suggests that the censorship infrastructure employed by the operator performs filtering by IP address to avoid having to perform Deep Packet Inspection (DPI) on all traffic to determine the domain name requested. Furthermore, the censorship infrastructure does not negotiate ECN. We did not investigate whether or not deep packet inspection was being performed.

Follow-up traceroute measurements to three targets via the EE network (Figure 3) show evidence of a TCP intercepting proxy with the first non-private [26] hop being the target web server. This was the same for websites with content and for censored websites. We also observed that attempts to visit HTTPS versions of websites that are censored for HTTP were successful. When conducting the same traceroute with the TCP destination port set to 443 (HTTPS), we saw the same intercepting proxy behavior. When using a random port (9283) we did not see the intercepting proxy and traceroute showed the path operated as expected. Traceroute functionality is planned for PATHspider with the possibility to in future automate traceroutes in response to anomalous replies.

We did not see any evidence that DNS was used as part of the censorship infrastructure (e.g., `pornhub.com` resolved to the same IP address within the EE network as from the Hetzner network).

## VIII. Discussion and Future Work

Previous studies with PATHspider used vantage points that revealed few middleboxes between them and the Internet. With our initial measurements in MONROE, we have, however, seen that the MBB environment is demonstrably far more complex. To get a better picture of the mobile ecosystem, we plan to further deploy a large-scale measurement campaign in MONROE, covering different operators in all the countries with MONROE presence.

As we proceed with a larger test campaign, we can compare the results we collect in MBB networks to the results we have collected from cloud vantage points to identify discrepancies. It will also be important to analyze responses from targets to identify censorship and other unexpected results, for example where a particular subscription has run out of data allowance and is served a captive portal page.

In the UK, Open Rights Group have built a probe for detecting censorship block pages[17] and we will use their patterns as a starting point to build up a set of patterns for all operators in the MONROE testbed. A similar repository of block pages is maintained by Citizen Lab[18].

In this study, we used a pre-resolved target list using DNS servers outside of the EE network. This means that the resolved target list cannot be influenced by DNS based censorship, but it also means that we will not have tested any content-distribution network edge caches within the EE network. It may be beneficial to run tests that involve lookups within the target network to include these in our target lists.

## IX. Conclusions

This paper has described the PATHspider measurement tool and its integration in the open access hardware-based MONROE platform. MONROE provides a wide area MBB testbed across which PATHspider can perform path transparency A/B tests. These measurements can indicate connectivity failure or other adverse effects when new transport techniques are introduced. As a proof-of-concept, we collected and analyzed an initial dataset from MONROE test nodes using the commercial

[17] https://www.blocked.org.uk/
[18] https://github.com/citizenlab/blockpages

EE operator in the UK. Our results have shown that the MBB environment is considerably different to the cloud vantage points that PATHspider has used in the past. In this particular MBB network, ECN was found safe for use, but could not be negotiated end-to-end and may instead be negotiated with a middlebox within the operator network. The developed tools will be used as a basis for a large-scale measurement campaign in MONROE to investigate whether new techniques are deployable in MBB networks across Europe or whether there is evidence that specific mechanisms may lead to adverse effects or connectivity failure. By learning more about support in MBB networks, developers of new protocol and innovations will be able to make more informed decisions about the design solutions.

## Acknowledgements

The MAMI project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 688421. This work was partially supported by the European Union's Horizon 2020 research and innovation program under grant agreement No. 644399 (MONROE). The opinions expressed and arguments employed reflect only the authors' views. The European Commission is not responsible for any use that may be made of that information.

## References

[1] Z. Wang, Z. Qian, Q. Xu, Z. M. Mao, and M. Zhang, "An untold story of middleboxes in cellular networks," in *Proceedings of ACM SIGCOMM*, 2011.

[2] G. Papastergiou, G. Fairhurst, D. Ros, A. Brunstrom, K.-J. Grinnemo, P. Hurtig, N. Khademi, M. Tüxen, M. Welzl, D. Damjanovic, and S. Mangiante, "De-ossifying the Internet transport layer: A survey and future perspectives," *IEEE Communications Surveys and Tutorials*, vol. 19, no. 1, pp. 619–639, 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7738442/

[3] J. Iyengar and M. Thomson, "QUIC: A UDP-Based Multiplexed and Secure Transport," Working Draft, IETF Secretariat, Internet-Draft draft-ietf-quic-transport-02, March 2017, http://www.ietf.org/internet-drafts/draft-ietf-quic-transport-02.txt. [Online]. Available: http://www.ietf.org/internet-drafts/draft-ietf-quic-transport-02.txt

[4] K. Moriarty and A. Morton, "Effect of pervasive encryption," Working Draft, IETF Secretariat, Internet-Draft draft-mm-wg-effect-encrypt-09, March 2017, http://www.ietf.org/internet-drafts/draft-mm-wg-effect-encrypt-09.txt. [Online]. Available: http://www.ietf.org/internet-drafts/draft-mm-wg-effect-encrypt-09.txt

[5] I. Learmonth, B. Trammell, M. Kühlewind, and G. Fairhurst, "PATHspider: A tool for active measurement of path transparency," in *First ACM/IRTF Applied Networking Research Workshop*, Berlin, Germany, Jul 2016.

[6] M. Kühlewind, S. Neuner, and B. Trammell, "On the state of ECN and TCP options on the Internet," in *Passive and Active Measurement Conference*, Hong Kong, China, 2013, pp. 135–144.

[7] B. Trammell, M. Kühlewind, D. Boppart, I. Learmonth, G. Fairhurst, and R. Scheffenegger, "Enabling Internet-Wide Deployment of Explicit Congestion Notification," in *Passive and Active Measurement Conference*, Brooklyn, USA, 2015, pp. 193–205.

[8] E. Gubser, "Explicit Congestion Negotiation (ECN) support based on P2P networks," ftp://ftp.tik.ee.ethz.ch/pub/students/2015-FS/SA-2015-05.pdf.

[9] K. Ramakrishnan, S. Floyd, and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP," RFC 3168 (Proposed Standard), Internet Engineering Task Force, Sep. 2001, updated by RFCs 4301, 6040. [Online]. Available: http://www.ietf.org/rfc/rfc3168.txt

[10] Y. Cheng, J. Chu, S. Radhakrishnan, and A. Jain, "TCP Fast Open," RFC 7413 (Experimental), Internet Engineering Task Force, Dec. 2014. [Online]. Available: http://www.ietf.org/rfc/rfc7413.txt

[11] K. Nichols, S. Blake, F. Baker, and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers," RFC 2474 (Proposed Standard), Internet Engineering Task Force, Dec. 1998, updated by RFCs 3168, 3260. [Online]. Available: http://www.ietf.org/rfc/rfc2474.txt

[12] Ö. Alay, A. Lutu, D. Ros, R. Garcia, V. Mancuso, A. F. Hansen, A. Brunstrom, M. A. Marsan, and H. Lonsethagen, "MONROE: Measuring mobile broadband networks in Europe," in *Proceedings of the IRTF & ISOC Workshop on Research and Applications of Internet Measurements (RAIM)*, 2015.

[13] Ö. Alay, A. Lutu, R. García, M. Peón-Quirós, V. Mancuso, T. Hirsch, T. Dely, J. Werme, K. Evensen, A. Hansen *et al.*, "Measuring and assessing mobile broadband networks with monroe," in *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2016 IEEE 17th International Symposium on A*. IEEE, 2016, pp. 1–3.

[14] R. N. Staff, "RIPE Atlas: A Global Internet Measurement Network," *Internet Protocol Journal*, vol. 18, no. 3, September 2015.

[15] A. Filasto and J. Appelbaum, "OONI: Open observatory of network interference." in *FOCI*, 2012.

[16] C. Kreibich, N. Weaver, B. Nechaev, and V. Paxson, "Netalyzr: Illuminating The Edge Network," in *Internet Measurement Conference (IMC)*, 2010.

[17] A. Faggiani, E. Gregori, L. Lenzini, S. Mainardi, and A. Vecchio, "On the feasibility of measuring the Internet through smartphone-based crowdsourcing," in *2012 10th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)*, May 2012, pp. 318–323.

[18] V. Thirion, K. Edeline, and B. Donnet, *Tracking Middleboxes in the Mobile World with TraceboxAndroid*. Cham: Springer International Publishing, 2015, pp. 79–91. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-17172-2_6

[19] G. Detal, B. Hesmans, O. Bonaventure, Y. Vanaubel, and B. Donnet, "Revealing Middlebox Interference with Tracebox," in *Proceedings of the 2013 Conference on Internet Measurement Conference*, ser. IMC '13. New York, NY, USA: ACM, 2013, pp. 1–8. [Online]. Available: http://doi.acm.org/10.1145/2504730.2504757

[20] S. Neuhaus, R. Müntener, K. Edeline, B. Donnet, and E. Gubser, "Towards an observatory for network transparency research," in *Proceedings of the 2016 Applied Networking Research Workshop*, ser. ANRW '16. New York, NY, USA: ACM, 2016, pp. 71–73. [Online]. Available: http://doi.acm.org/10.1145/2959424.2959425

[21] M. Belshe, R. Peon, and M. Thomson, "Hypertext Transfer Protocol Version 2 (HTTP/2)," RFC 7540 (Proposed Standard), Internet Engineering Task Force, May 2015. [Online]. Available: http://www.ietf.org/rfc/rfc7540.txt

[22] S. Friedl, A. Popov, A. Langley, and E. Stephan, "Transport Layer Security (TLS) Application-Layer Protocol Negotiation Extension," RFC 7301 (Proposed Standard), Internet Engineering Task Force, Jul. 2014. [Online]. Available: http://www.ietf.org/rfc/rfc7301.txt

[23] B. Trammell and M. Kuehlewind, "Path layer udp substrate specification," Working Draft, IETF Secretariat, Internet-Draft draft-trammell-plus-spec-01, March 2017, http://www.ietf.org/internet-drafts/draft-trammell-plus-spec-01.txt. [Online]. Available: http://www.ietf.org/internet-drafts/draft-trammell-plus-spec-01.txt

[24] M. Varvello, K. Schomp, D. Naylor, J. Blackburn, A. Finamore, and K. Papagiannaki, "Is The Web HTTP/2 Yet?" in *Passive and Active Measurements (PAM '16)*, 2016.

[25] C. Davies, "Broadband firms urged to block sex websites to protect children," *The Guardian*, Dec. 2010. [Online]. Available: https://www.theguardian.com/society/2010/dec/19/broadband-sex-safeguard-children-vaizey

[26] Y. Rekhter, B. Moskowitz, D. Karrenberg, G. J. de Groot, and E. Lear, "Address Allocation for Private Internets," RFC 1918 (Best Current Practice), Internet Engineering Task Force, Feb. 1996, updated by RFC 6761. [Online]. Available: http://www.ietf.org/rfc/rfc1918.txt