Leveraging the 0-RTT Convert Protocol to improve Wi-Fi/Cellular convergence

Matthieu Baerts
Tessares
Belgium
Matthieu.Baerts@tessares.net

Nicolas Keukeleire
Tessares
Belgium
Nicolas.Keukeleire@tessares.net

Olivier Bonaventure
Tessares
Belgium
Olivier.Bonaventure@tessares.net

ABSTRACT

Mobile data usage is increasing and mobile network operators are looking to move some of this traffic onto Wi-Fi networks. MPTCP and the 0-RTT TCP Convert protocol can help these operators by enabling a convergence of cellular and Wi-Fi networks. We first present these two protocols, then discuss possible deployments and finally show some measurements from a smartphone.

CCS CONCEPTS

• **Networks** → *Network components*; **Transport protocols**.

ACM Reference Format:
Matthieu Baerts, Nicolas Keukeleire, and Olivier Bonaventure. 2021. Leveraging the 0-RTT Convert Protocol to improve Wi-Fi/Cellular convergence. In *Applied Networking Research Workshop (ANRW '21), July 24–30, 2021, Virtual Event, USA.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3472305.3472317

1 INTRODUCTION

During the last decade, the amount of data exchanged by mobile – mainly smartphone and tablet – users has grown tremendously[13, 14]. Industry analysts indicate that more than one-third of the cellular network operators report more than 10 GBytes of data per SIM per month [2]. This growth is mainly driven by users on unlimited data plans. During the last year, worldwide cellular data consumption grew by 38%, but the cellular operators' revenues decreased by 2%. This puts cellular operators under pressure. To cope with this growth, cellular network operators are deploying more antennas and new technologies such as 5G, but it will be difficult for them to increase their pricing to recover these additional costs. On the other hand, smartphones and tablets can be attached to both Wi-Fi and cellular networks. End users on unlimited data plans do not know or care whether their data is received over Wi-Fi or cellular.

Many cellular network operators and mobile virtual network operators also provide fixed Internet access services, and as a consequence provide Internet access using Wi-Fi in hundreds of thousands of households, enterprise networks, and public hotspots. Various solutions have been proposed and deployed to make it simpler for users to access these Wi-Fi networks, e.g. FON, Boingo, Eduroam for universities [21] and more recently the OpenRoaming initiative. With all these initiatives, cellular network operators want to offload as much traffic as possible over these managed Wi-Fi networks. In this short paper, we describe how deployments can leverage MPTCP [15] and the 0-RTT Convert protocol [4].

2 THE 0-RTT TCP CONVERT PROTOCOL

Multipath TCP (MPTCP) [15] is a TCP extension which allows one TCP connection to be conveyed on several paths – several TCP subflows – over different access networks. MPTCP is used in Korea to combine Wi-Fi and cellular [19], on all Apple's iPhones to improve quality of experience for Siri, Maps, and Music [3, 5] and for Hybrid Access Networks [17].

The 0-RTT Convert protocol [4] was designed to ease the deployment of TCP extensions such as MPTCP. Ideally, MPTCP should be used on both clients and end-servers. However, since most of the benefits of MPTCP come from the ability to share different access links, another deployment option is to leverage MPTCP-enabled proxies. These proxies terminate the MPTCP connections created by clients and use TCP to interact with the distant servers. When using the 0-RTT protocol, the client creates an MPTCP connection to such a proxy called a 0-RTT Converter. This 0-RTT Converter uses a dedicated IP address and port to accept the incoming connection. Having created the connection the client immediately sends a 0-RTT Convert message that specifies the IP address and port number of the remote server. The 0-RTT Converter creates a TCP connection towards the specified server address and port, and relays all data exchanged over the two proxied connections. To minimize the latency impact of the 0-RTT Converter, the 0-RTT Convert protocol

ANRW '21, July 24–30, 2021, Virtual Event, USA

ACM ISBN 978-1-4503-8618-0/21/07.
https://doi.org/10.1145/3472305.3472317

leverages TCP Fast Open [7] to exchange the address and port of the remote server during the connection's handshake. When the 0-RTT Converter-initiated connection succeeds, the 0-RTT Converter informs the client of the TCP options that are supported by the server. In case of issues with TCP Fast Open, a fall back on regular 3WHS is done. Two main use cases have been defined for the 0-RTT protocol. First, the Broadband Forum has adopted this protocol to support hybrid access networks that combine xDSL and cellular access networks [16]. Second, 3GPP has adopted this protocol to support Wi-Fi/cellular convergence in 5G networks [1, 8] by introducing a new feature named Access Traffic Steering, Switching and Splitting (ATSSS).

3 DEPLOYMENT AND MEASUREMENTS

There are two main possible deployments to provide Wi-Fi/cellular convergence using MPTCP and the 0-RTT Convert protocol. First, cellular network operators that also manage a large number of Wi-Fi access points can deploy 0-RTT Converter servers in their backbone where their fixed and cellular networks converge. The 0-RTT Converter would enable them to not only support seamless handovers for all TCP-based applications but also provide bandwidth aggregation when needed, prioritizing the Wi-Fi network when it provides similar or better performance than the cellular one. In the long term, it can be expected that these deployments will migrate to use the 3GPP ATSSS feature in a 5G Core. Secondly, large enterprises having campuses with Wi-Fi and private 4G/5G could also provide similar services by using 0-RTT Converters as part of the Multi-access Edge Computing (MEC) platform of the service provider [12].

To assess the benefits of the 0-RTT Convert protocol, we have developed the `libconvert` [20] open-source library that implements the client-side 0-RTT Convert protocol. Thanks to this library, only around 250 lines were needed to support the protocol in `shadowsocks-libev`'s C code [11]. New options have been added to the ShadowSocks Android app [10]. We also ported the off-tree implementation of MPTCP to the 4.9 Linux kernel used by the Xiaomi Pocophone F1 with Android 9 [9]. The 0-RTT Converter is an extension of the Hybrid Access Gateway (HAG) used for hybrid (xDSL+4G) networks [17].

We measured handover scenarios for this lightning paper. On the smartphone, we use the Twitch application to watch live video content sent over TCP. The smartphone is initially connected to two distinct networks: a public LTE network and a private home Wi-Fi access point. The 0-RTT Converter server is hosted on a public cloud and is configured to prioritize sending data over the Wi-Fi network. Our Wi-Fi network is limited to 10 Mbps due to the underlying DSL bandwidth.

Figure 1 first describes the setup and then shows how the smartphone uses the networks while using Twitch. The blue

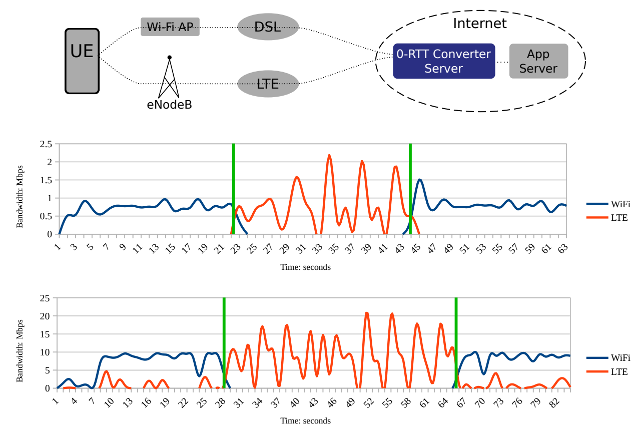

Figure 1: MPTCP and the 0-RTT Convert protocol deployed on a User Equipment (top) enable seamless handovers (middle) and bandwidth aggregation (bottom) while prioritizing Wi-Fi over LTE

line is the Wi-Fi usage while the red line shows the LTE usage. The LTE is from a commercial network from a local operator, hence the variability of the provided bandwidth. The green bars indicate when the Wi-Fi stops or comes back. The middle part of Figure 1 shows a scenario where Wi-Fi is prioritized. All the data is exchanged over Wi-Fi when this network is active. After 22 seconds, the smartphone is disconnected from the Wi-Fi but the connection seamlessly continues over the LTE network. When the Wi-Fi network comes back, the connection continues over Wi-Fi. Without MPTCP and the 0-RTT Converter, the connections break and the application has to restart them over the LTE network when available: a seamless handover is not possible. The bottom part of Figure 1 shows a scenario where the smartphone uses the LTE network to improve the quality of experience. During the first 28 seconds, the Wi-Fi network is not sufficient to carry the entire Twitch stream and some data is transmitted over the LTE network. Without MPTCP and the 0-RTT Converter, it would not have been possible to watch the video at this quality without interruptions. Then the Wi-Fi fails and the connection continues over the LTE network.

4 CONCLUSION

3GPP has selected MPTCP and the 0-RTT Convert Protocol to support Access Traffic Steering, Switching & Splitting for TCP-based applications. We have demonstrated the first implementation of the 0-RTT Convert Protocol. Our measurements indicate that this solution works well in existing 4G and Wi-Fi networks and could already be deployed in existing networks without waiting for the roll-out of 5G networks that include ATSSS. In the longer term, we expect to see an evolution based on QUIC or Multipath QUIC with tunneling [6, 18] which would support non-TCP traffic.

ACKNOWLEDGEMENTS

This work is supported by NGI POINTER (NGI Program for Open Internet Renovation) which has received funding from the European Commission, as part of the Horizon 2020 Research and Innovation Programme, under Grant Agreement N°871528.

We would like to thank our colleague Graham Turnbull and the anonymous reviewers for their helpful comments and suggestions to improve and clarify this short paper.

REFERENCES

[1] 3GPP (3rd Generation Partnership Project). 2019. Technical Specification Group Services and System Aspects; System Architecture for the 5G System; Stage 2 (Release 16). TS23501. https://www.3gpp.org/ftp/Specs/archive/23_series/23.501/

[2] Tefficient AB. 2021. 2020 was a horrific year – but not for the global mobile data traffic that grew 38%. https://tefficient.com/wp-content/uploads/2021/04/tefficient-industry-analysis-1-2021-mobile-data-usage-and-revenue-FY-2020-per-operator-13-April-2021.pdf.

[3] Olivier Bonaventure. 2019. Apple Music on iOS13 uses Multipath TCP through load-balancers. http://blog.multipath-tcp.org/blog/html/2019/10/27/apple_music_on_ios13_uses_multipath_tcp_through_load_balancers.html.

[4] Olivier Bonaventure, Mohamed Boucadair, Sri Gundavelli, SungHoon Seo, and Benjamin Hesmans. 2020. 0-RTT TCP Convert Protocol. RFC 8803. https://doi.org/10.17487/RFC8803

[5] Olivier Bonaventure and S Seo. 2016. Multipath TCP deployments. *IETF Journal* 12, 2 (2016), 24–27.

[6] Mohamed Boucadair, Olivier Bonaventure, Maxime Piraux, Quentin De Coninck, Spencer Dawkins, Mirja Kühlewind, Markus Amend, Andreas Kassler, Qing An, Nicolas Keukeleire, and SungHoon Seo. 2020. *3GPP Access Traffic Steering Switching and Splitting (ATSSS) - Overview for IETF Participants*. Internet-Draft draft-bonaventure-quic-atsss-overview-00. Internet Engineering Task Force. https://datatracker.ietf.org/doc/html/draft-bonaventure-quic-atsss-overview-00 Work in Progress.

[7] Yuchung Cheng, Jerry Chu, Sivasankar Radhakrishnan, and Arvind Jain. 2014. TCP Fast Open. RFC 7413. https://doi.org/10.17487/RFC7413

[8] Massimo Condoluci, Stephen H Johnson, Vicknesan Ayadurai, Maria A Lema, Maria A Cuevas, Mischa Dohler, and Toktam Mahmoodi. 2019. Fixed-mobile convergence in the 5G era: From hybrid access to converged core. *IEEE Network* 33, 2 (2019), 138–145.

[9] Linux Kernel Developers. 2021. Linux 4.9 Android kernel supporting MPTCP and suitable for the Xiaomi Pocophone F1. https://github.com/Tessares/android_kernel_xiaomi_sdm845.

[10] ShadowSocks Developers. 2021. ShadowSocks Android application with `libconvert` support. https://github.com/Tessares/shadowsocks-android.

[11] ShadowSocks Developers. 2021. ShadowSocks library with `libconvert` support. https://github.com/Tessares/shadowsocks-libev.

[12] George Elissaios. 2021. AWS and Verizon Expand 5G Collaboration with Private MEC Solution. https://aws.amazon.com/blogs/industries/aws-and-verizon-expand-5g-collaboration-with-private-mec-solution/.

[13] Ericsson. 2014. Ericsson Mobility Report. (Nov 2014). https://www.ericsson.com/49de8a/assets/local/mobility-report/documents/2014/ericsson-mobility-report-november-2014.pdf

[14] Ericsson. 2020. Ericsson Mobility Report. (Nov 2020). https://www.ericsson.com/4adc87/assets/local/mobility-report/documents/2020/november-2020-ericsson-mobility-report.pdf

[15] Alan Ford, Costin Raiciu, Mark J. Handley, and Olivier Bonaventure. 2013. TCP Extensions for Multipath Operation with Multiple Addresses. RFC 6824. https://doi.org/10.17487/RFC6824

[16] Broadband Forum. 2019. Nodal Requirements for Hybrid Access Broadband Networks. WT-378.

[17] Nicolas Keukeleire, Benjamin Hesmans, and Olivier Bonaventure. 2020. Increasing Broadband Reach with Hybrid Access Networks. *IEEE Communications Standards Magazine* 4, 1 (2020), 43–49.

[18] David Schinazi. 2021. *The MASQUE Protocol*. Internet-Draft draft-schinazi-masque-protocol-03. Internet Engineering Task Force. https://datatracker.ietf.org/doc/html/draft-schinazi-masque-protocol-03 Work in Progress.

[19] S Seo. 2015. Kt's giga lte. *Presentation at IETF* 93 (2015).

[20] Tessares. 2021. `libconvert`, a client-side library implementing the 0-RTT Convert Protocol. https://github.com/Tessares/libconvert.

[21] Klaas Wierenga, Stefan Winter, and Tomasz Wolniewicz. 2015. The eduroam Architecture for Network Roaming. RFC 7593. https://doi.org/10.17487/RFC7593