Evaluation and Prototyping of Multipath Protocol Mechanisms

Michael Scharf, Thomas-Rolf Banniza, Peter Schefczik
Alcatel-Lucent Bell Labs
{firstname.lastname}@alcatel-lucent.com

Amanpreet Singh, Andreas Timm-Giel
University of Bremen, Hamburg University of Technology
aps@comnets.uni-bremen.de, timm-giel@tuhh.de

I. Introduction

Multipath transport is a promising new Internet paradigm. Most Internet applications use the Transmission Control Protocol (TCP), which cannot efficiently use multiple available paths between the connection endpoints. As a consequence, there is a significant potential for new protocol mechanisms that overcome this limitation of the current Internet.

There are several situations in which data transport over multiple paths would make sense: End users access the Internet more and more from mobile devices that have multiple network interfaces, which could be used in parallel to use their aggregated bandwidth. Also, data centers are typically connected to the Internet by several uplinks in order to improve the reliability and to enable load balancing (see Fig. 1). The latter use case is particularly important for network-based computing applications that are realized in the cloud, that have high performance and reliability requirements, and that are the focus of the G-Lab NETCOMP project [1].

Multipath transport is not an entirely new concept and has been researched before. Still, there are new efforts to develop multipath transport protocols that could be used at Internet scale [2]. The Internet Engineering Task Force (IETF) has started corresponding standardization activities [3].

This abstract describes work-in-progress that develops, evaluates, and prototypes multipath protocols for NETCOMP applications. In Section II we first survey and classify multipath solutions and identify open issues that require further work. Two examples are the design of interfaces towards applications [4], and the cross-layer interaction between transport and routing. Section III presents the current status of our prototyping activities for new multipath protocol mechanisms. Section IV briefly summarizes some lessons learnt from initial measurements. Finally, this document concludes with a summary and an outlook.

II. Multipath Design Space

A. Classification of Approaches

Data can be transported over multiple paths by different mechanisms: On the one hand, routing protocols decide how traffic is forwarded in a network. *Multipath routing protocols* can make use of several available paths between the same ingress and egress gateways, e. g., for load balancing. But, on the other hand, hosts can have multiple interfaces with different addresses, and then functions in the end-system are needed that manage the interfaces and deal with their different and usually variable characteristics. In the latter case, the multipath functionality can either be realized in the application (*multipath-aware application*), or the host can deal with this in the network stack using a *multipath transport protocol*.

Multipath transport and routing are not independent: Routing schemes are typically unaware of flows, and this can cause problems (e. g., out-of-order packets). In contrast, transport protocols do not know about network topologies, e. g., whether paths are disjoint. Therefore, multipath routing and transport could benefit from *cross-layer information exchange*: Routing protocols could expose available routes, and transport protocols could provide information about the traffic. However, this would require enhancements of the Internet architecture.

B. A New Solution – IETF Multipath TCP

Recently, the IETF started to develop Multipath TCP (MPTCP). This TCP extension adds the capability of using multiple paths to a regular TCP session in order to increase the throughput, the overall resource utilization, and the resilience to network failure [3]. MPTCP is expected to offer the same reliable, in-order, byte-stream transport as TCP, and is designed to be backward-compatible. There are different possible realizations of the protocol, such as a protocol for end-systems with multiple addresses [5]. Since multipath transport can significantly affect resource sharing, new congestion control algorithms are required, too. One example is [6].

C. Identified Open Issues

MPTCP is a promising first step to enable multipath transport in the Internet. Yet, there are several open issues that require further research. One focus of our ongoing research is the application interface (API): Unlike a clean-slate approach, MPTCP must be usable with the large variety of applications that already exist in the Internet [4]. The API must be carefully designed so that MPTCP-unaware applications cannot break, which means that the compatibility with legacy applications must be tested.

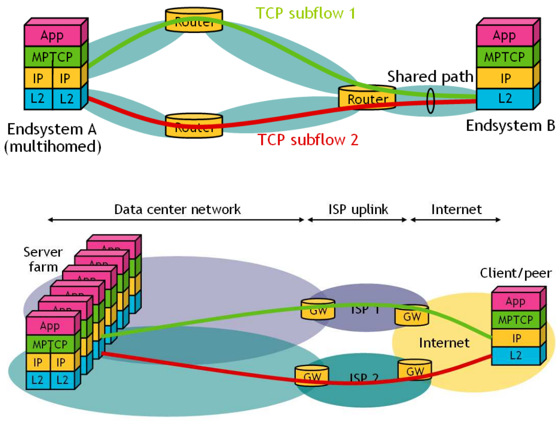

Figure 1: Multipath transport in general and in datacenter networks

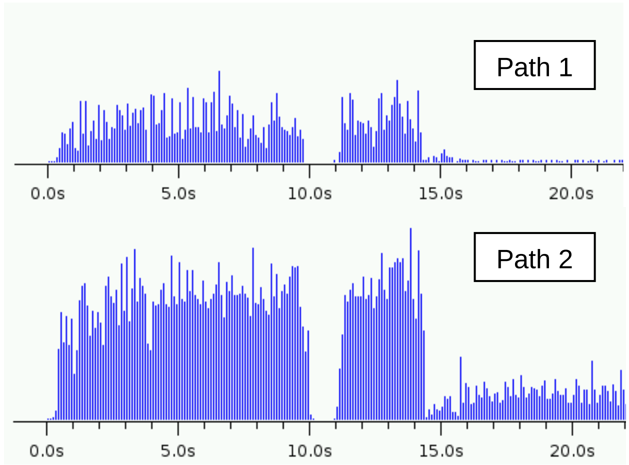

Figure 2: Bandwidth sharing measurement of user-space MPTCP

In addition to this, the interaction of MPTCP and routing is not well understood so far. A new generic, cross-layer signaling solution is needed and could significantly help to optimize multipath transport.

Finally, there are numerous algorithmic problems, such as congestion control mechanisms other than the one in [6].

III. Ongoing Development and Prototyping Work

A. Existing Prototypes

Enhanced multipath protocol mechanisms can only be tested in experiments under realistic constraints. This is why prototype implementations are required. Concerning MPTCP, two implementations already exist, a patch for the Linux kernel [7] and a user-space implementation [8]. We have studied both implementations concerning features, maturity, and extensibility. Both implementations are still very incomplete and it is therefore difficult to use them as a basis for own protocol enhancements. Some example results of our evaluation are presented in the next section.

B. Ongoing Development Work

The real-world deployment of any multipath mechanism would significantly benefit from a simple user-space solution. Also, user-space solutions are much better suited for tests in experimental facilities. This is why we are working on a new multipath transport implementation that realizes most functionality in the user space. It will be extensible so that new protocol mechanisms and algorithms can be added in a later stage. Our prototype is work-in-progress. The final poster will present further details about the structure and the current status.

IV. Initial Evaluation Results

We performed several experiments with available MPTCP implementations, using a laboratory setup with several virtual machines that are interconnected by an emulated topology. Fig. 2 shows, as an example, the bandwidth used by the user-space MPTCP implementation [8]. In the assumed topology, the connection endpoints can use two equal paths in parallel. The plots in Fig. 2 show that, as to be expected, traffic is indeed distributed over the different paths. However, the load is not equally shared on both paths, and there are further unexpected effects. This illustrates the need for further development work on MPTCP implementations and also raises questions concerning the algorithms used therein, e. g., the congestion control mechanisms.

V. Summary and Outlook

The objective of this work is an enhanced and extensible multipath protocol solution based on IETF MPTCP. This document briefly summaries our ongoing work towards that goal. We classify the different design possibilities for multipath transport as well as open issues. This is the starting point for our conceptual work that in particular studies interfaces. Our corresponding prototype is still work-in-progress, given that the existing code basis is not sufficient for our purposes.

At the moment, tests and measurements are performed in laboratory setups. Once our prototype code becomes more stable, it is foreseen to perform larger-scale, realistic experiments in the G-Lab experimental facilities.

References

[1] BMBF Förderkennzeichen 01BK09

[2] J. Arkko, B. Briscoe, L. Eggert, A. Feldmann, and M. Handley, Dagstuhl perspectives workshop on end-to-end protocols for the future internet, ACM SIGCOMM Comput. Commun. Rev. 39, 2, pp. 42-47, Mar. 2009

[3] A. Ford, C. Raiciu, S. Barre, and J. Iyengar, Architectural Guidelines for Multipath TCP Development, draft-ietf-mptcp-architecture-00 (work in progress), Mar. 2010

[4] M. Scharf, and A. Ford, MPTCP Application Interface Considerations, draft-scharf-mptcp-api-01 (work in progress), Mar. 2010

[5] A. Ford, C. Raiciu, and M. Handley, TCP Extensions for Multipath Operation with Multiple Addresses, draft-ford-mptcp-multiaddressed-03 (work in progress), Mar. 2010

[6] C. Raiciu, M. Handley, and D. Wischik, Coupled Multipath-Aware Congestion Control, draft-raiciu-mptcp-congestion-01 (work in progress), Mar. 2010

[7] S. Barré, Linux MPTCP version 0.3, available from the Web site http://inl.info.ucl.ac.be/mptcp, 2010

[8] C. Raiciu, D. Wischik, and M. Handley, Practical Congestion Control for Multipath Transport Protocols, UCL Technical Report, 2009