ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Prof. Laurent Vanbever
Networked Systems Group

On combining SWIFT and Blink to improve the Internet convergence time

Master thesis proposal

BGP, the de-facto inter-domain routing protocol, converges slowly upon failures [9, 11]. In addition, failures may affect a large number of prefixes, and the time it takes for a router to update the data plane increases linearly with the number of prefixes to update [3]. Putting everything together, an Internet failure can result in minutes of downtime — enough to potentially violate the SLAs of an ISP and degrade its reputation. Many fast reroute systems have been designed in the past to speed up the convergence, but they only focus on local failures [4, 7], *i.e.,* the ones occurring in the ISP's network, or require to change the protocols [10].

Fortunately, our group has recently developed systems to quickly detect local *and* remote Internet failures and reroute traffic to restore connectivity. The first one is **SWIFT** [9], a system that focuses on large Internet failures affecting thousands of prefixes. To speed up convergence, SWIFT predicts the extent of a remote failure from a few received BGP updates, leveraging the fact that such updates are correlated (*e.g.,* they share the same AS path), and then reroutes the traffic for all the affected prefixes with just few data-plane updates that match on data-plane tags embedded in each data packet. The fundamental problem with SWIFT though, is that it can take minutes for the *first* BGP update to propagate after the corresponding failure data-plane failure. To answer this problem, we then designed **Blink** [8], a system that detects Internet failures and reroutes traffic entirely in the data plane using the recent programmable switches [5, 6] (such as Tofino [1]). Blink is able to detect a failure within few seconds by looking at the TCP retransmissions of the TCP flows, and can immediately reroute traffic, at line rate. But because Blink runs in the data plane, it has limited resources and only uses simple algorithms (*e.g.,* a static threshold to infer failures) that run on a per-prefix basis. Consequently, Blink only works for complete failures (*i.e.,* the ones affecting all the flows destined to a prefix), requires some extra time to recover connectivity if the first back next-hop is also affected by the failure, and only works for a limited number of prefixes (up to 10k prefixes while there are more than 700k advertised prefixes nowadays [2]).

In this thesis, the goal is to combine some of the mechanisms used in SWIFT and Blink in order to build a new fast reroute system that solves the aforementioned problems. To be fast, the new system will rely on data plane signals (like in Blink), and will then run the main algorithms in the control plane so as to have more resources to run advanced algorithms (such as the ones used in SWIFT). This design can be achieved simply by mirroring the TCP flows that Blink monitors to a controller which then runs the advanced algorithms and triggers the rerouting when it detects a failure.

During the thesis, the student is expected to look at how information from multiple prefixes can be combined to improve the rerouting. For example, combining information from multiple prefixes could allow the system to localize a failure so as to reroute around it thanks to a data-plane encoding. This is similar to SWIFT, but instead of the BGP updates, the input is now the data-plane signals (TCP retransmissions). Combining multiple prefixes can also strengthen the signal and improve the accuracy of the system, in particular for partial failures where the per-prefix signals may be to low to infer any failure.

Milestones

- Understand SWIFT, Blink, and their pros and cons;
- Study how can we improve the rerouting by combining the signals from multiple prefixes, and propose new algorithms to do that.
- Implement the proposed algorithms fully in Python (or a similar language) and evaluate them on real and synthetic traces.
- Implement the end-to-end system, with its P4 part (data plane) and Python part (control plane).
- Finally, test the system on a Tofino switch and a dedicated server that we have in our lab.

Prerequisites

- Being able to program in Python, good knowledge in UNIX-like systems, some knowledge in P4 is a plus;
- Communication Networks (227-0120-00L), or equivalents.

Contact

- Thomas Holterbach, thomahol@ethz.ch
- Prof. Dr. Laurent Vanbever, lvanbever@ethz.ch

References

[1] Barefoot tofino, the world's fastest p4-programmable ethernet switch asics. https://barefootnetworks.com/products/brief-tofino/.

[2] Cidr report. https://www.cidr-report.org/as2.0/.

[3] M. Alan Chang, T. Holterbach, M. Happe, and L. Vanbever. Supercharge me: Boost router convergence with sdn. In *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*, SIGCOMM '15, pages 341–342, New York, NY, USA, 2015. ACM.

[4] A. Atlas and A. Zinin. Basic Specification for IP Fast Reroute: Loop-Free Alternates. RFC 5286, Sept. 2008.

[5] P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese, and D. Walker. P4: Programming protocol-independent packet processors. *SIGCOMM Comput. Commun. Rev.*, 44(3):87–95, July 2014.

[6] P. Bosshart, G. Gibb, H.-S. Kim, G. Varghese, N. McKeown, M. Izzard, F. Mujica, and M. Horowitz. Forwarding metamorphosis: Fast programmable match-action processing in hardware for sdn. In *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM*, SIGCOMM '13, pages 99–110, New York, NY, USA, 2013. ACM.

[7] C. Filsfils, P. Mohapatra, J. Bettink, P. Dharwadkar, P. D. Vriendt, Y. Tsier, V. V. D. Schrieck, O. Bonaventure, and P. Francois. BGP Prefix Independent Convergence (PIC) Technical Report. Technical report, Cisco, 2011.

[8] T. Holterbach, E. C. Molero, M. Apostolaki, A. Dainotti, S. Vissicchio, and L. Vanbever. Blink: Fast connectivity recovery entirely in the data plane. In *16th USENIX Symposium on Networked Systems Design and Implementation (NSDI 19)*, pages 161–176, Boston, MA, 2019. USENIX Association.

[9] T. Holterbach, S. Vissicchio, A. Dainotti, and L. Vanbever. Swift: Predictive fast reroute. In *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*, SIGCOMM '17, pages 460–473, New York, NY, USA, 2017. ACM.

[10] N. Kushman, S. Kandula, D. Katabi, and B. M. Maggs. R-BGP: Staying connected in a connected world. USENIX, 2007.

[11] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian. Delayed internet routing convergence. In *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, SIGCOMM '00, pages 175–187, New York, NY, USA, 2000. ACM.