Take the mini-Internet to the next level

Semester thesis proposal

One practical group project in our communication networks lecture is the routing project [1]. For this project, we build a virtual mini-Internet infrastructure composed of hundreds of routers and dozens of Autonomous Systems (ASes) in one of our server and let the students configure their ASes and virtual devices. They have to configure various routing protocols to enable Internet-wide connectivity, for instance OSPF to enable intra-domain connectivity and BGP to enable connectivity between different ASes.

Motivated by the positive feedback we received from the students following our lecture, we decided to make the platform publicly available [2,6]. Several universities and companies have already started to use the platform. As a result, we plan to add new features and improve the existing code. The goal of this thesis is to implement one new functionality. Following we describe three different directions we would like to explore:

**Implementing the RPKI infrastructure in the mini-Internet.** The Resource Public Key Infrastructure (RPKI) started to be developed in 2008 and aims at preventing BGP hijacks [4]. Since 2011, it has been more and more deployed within the Internet [5,7]. Given the high interest of the network community, we decided to implement the RPKI infrastructure within the mini-Internet. The goal of this thesis is to continue the work done during a previous semester thesis [3]. More precisely, the student is expected to improve how the RPKI infrastructure is implemented to better reflect the actual implementation in the real Internet. The student is also expected to implement a web-interface so that students can graphically issue and manipulate Route Origin Authorisations (ROAs).

**Improving the configuration and testing of the platform.** As the mini-Internet continues to grow, we continuously add new changes and features. Making sure that configuring the platform remains easy and that all the pieces work well regardless of how the platform is configured gets more and more complicated. For example, for each new feature we have to make sure that the mini-Internet builds correctly, that basic properties (such as reachability between hosts) hold and that we can clean it up safely without affecting other processes running in the server. To address these challenges, the student is first expected to study which Internet properties should always hold given a specific input network (e.g., reachability), and which type of tests we would need to perform to make sure that the mini-Internet is working properly (e.g., unit tests, integration tests, stress tests). Afterwards, the student is expected to add a test framework to the existing mini-Internet code.

**Deploying a mini-Internet instance across multiple servers.** As of now the platform builds a mini-Internet in a single server. The number of virtual switches, routers and hosts is thus limited by the capacity of the server (especially CPU cores and available memory). For instance, a mini-Internet with 60 ASes, each composed of 8 routers and 4 switches, uses 58% of the memory and 51% of the CPU cores on a server with a total of 24 CPU cores and 256GB of RAM [2]. If the number of students increases or if one requires a larger mini-Internet (e.g., for research purposes), the capacity of a single server may become a limitation. The goal of this thesis is to improve the platform so that a single mini-Internet instance can run across multiple servers. The size of the mini-Internet should then be limited by the overall amount of CPU cores and memory across all the servers. The way the mini-Internet is divided across multiple servers should also be configurable.

If you want to help us to further develop the mini-Internet platform, we would be happy to talk with you and see which direction we could take based on your preference and skills.
Prerequisites

• Being able to program in Bash or Python, good knowledge in UNIX-like systems;
• Basic knowledge of virtualization/containerization (e.g., Docker);
• Basic knowledge in cryptography and/or network security (for the RPKI proposal);
• Communication Networks (227-0120-00L), or equivalents.

Contact

• Thomas Holterbach, thomahol@ethz.ch
• Tobias Bühler, buehlert@ethz.ch
• Prof. Dr. Laurent Vanbever, lvanbever@ethz.ch

References