Setting up a secure multicontroller SDN architecture

I. Context

Today's networks are managed by administrators based on metrological and monitoring information to react to changes in traffic and communication infrastructure failures. A possible architecture consists in automating this network management by deploying network controllers, dynamically capable of detecting and even anticipating the occurrence of changes in network system states and programming its reconfiguration as in the Software-Defined Networking (SDN) architecture. This centralized approach is increasingly used by companies such as Google, the world's number 1 network, which has part of its infrastructure under SDN.

SDN controllers are solely responsible for controlling the network. As a result, service on the network is degraded (even blocked) in the event of controller **failure or attack**. From a security point of view, it seems necessary to consider architectures with multiple controllers, thus avoiding the single point of failure/invasion and ensuring redundancy. Since even if a controller is only in charge of a subnetwork, it must have access to all network information to make decisions. Hence it is necessary to introduce a **communication interface between** controllers. This interface (named East-West) can be insecure, requires active maintenance of the connection, and **does not benefit from recognized standardized protocols**. Thus, an attack can be spread though this interface by the transmission of bad information in the same way as a BGP hijack.

II. Objective

The objective is to propose a reliable control architecture to guarantee a certain level of quality of service on the network, while maintaining a high level of security. As a first step, a multi-controller architecture will have to be considered. This is essential if the architecture is to be scalable. The goal is to secure the controller decision-making to limit the impact of an attacker by setting up a **consensus** (PBFT) between the various controllers.

The problem of implementing consensus in a multi-controller control architecture is to strike a balance between reactivity and security. Indeed, a consensus may take time to reach and PBFT has a complexity in $O(n^2)$ where n is the number of consensus participants. This time is a latency for setting up the data plane and has a direct impact on network **quality of service (QoS)**. Thus, increasing the number n of consensus participants complicates the task of attackers, but increases latency between controllers and thus on the network. The originality of the intership lies in the search for a compromise on the number of controllers participating in the consensus, which is essential for scaling up, to guarantee both a correct level of security and a correct level of quality of service. In our opinion, this compromise can be established either **analytically** or **empirically**, based on experiments on Grid'5000. The sensitivity of this compromise can be studied, depending on the consensus algorithm used.

Place of the internship: LIP, ENS de Lyon

Supervisors: Loïc Desgeorges; loic.desgeorges@univ-lyon1.fr

Skills required: network skills (TCP/IP stack, network performance, notion of QoS). Consensus knowledge is a plus. Fluency in English.

Starting date: between the beginning of February and the end of March 2024