Securing Network Applications in Software Defined Networks

Mots clés :
- Directeur de thèse : Farid NAIT-ABDESSELAM
- Co-encadrant(s) :
- Unité de recherche : Laboratoire d'Informatique Paris Descartes
- Ecole doctorale : École Doctorale Informatique, Télécommunications, Électronique de Paris
- Domaine scientifique principal : Divers

Résumé du projet de recherche (Langue 1)

As a result of various Internet services, computer networks has been recognized as a critical role in modern life over the past half century. The rapid development and convergence of computing technologies and communication create the need to connect diverse devices with different operating systems and protocols. This results in numerous challenges to provide seamless integration of a large amount of heterogeneous physical devices or entities. Hence, Software-defined networks (SDN) as an emerging paradigm has the potential to revolutionize legacy network management by centralizing the control and global visibility over the whole network. However, security issues remain a significant concern and impede SDN from being widely adopted. To identity the threats, we conducted an analysis in 3 dimensions to evaluate the security of SDN. In this analysis, we resumed 9 security principles for the SDN controller and checked the security of the current SDN controllers with these principles. We found that the SDN controllers, ONOS and OpenContrail are relatively more secure than the others according to our analysis methodology. We also found the urgent need to mitigate the problem of malicious network injection. Hence, we proposed a security-enhancing layer (SE-layer) to protect the interaction between the control plane and the application plane. This SE-layer is controller-independent and can work with OpenDaylight, ONOS, Floodlight, Ryu and POX, with low deployment complexity. No modification of their source codes is required in their implementation while the overall security of the SDN controller is enhanced. The prototype I, Controller SEPA, protects the SDN controller with network application authentication, authorization, application isolation, and information shielding with negligible latency from less than 0.1% to 0.3%. We developed the SE-layer prototype II, called Controller DAC, which makes the access control dynamic. Controller DAC can detect the API abuse by accounting the network application operation with latency less than 0.5%. Thanks to this SE-layer, the overall security of the SDN controller is improved but with a latency of less than 0.5%. Furthermore, we attempted to provide a secure network application deployment framework for the SDN controller with an orchestrator. First, we secured the SDN controller by using message queue to replace the current popular northbound interfaces, including RESTful APIs and native internal APIs, with a decomposable event-driven northbound interface. With this novel northbound interface, the orchestrator can deploy the network applications in the sandbox with resource control and access control. This approach can effectively shield against threats, which include resource exhaustion attacks and data tempering on the current SDN controller. We also implemented a network application deployed by the orchestrator to detect an OpenFlow specific attack, called priority-bypassing attack, for evaluating the utility of this northbound interface. In long term, the processing time for a packet_in message in this northbound interface is less than five milliseconds but the network application can be completely decoupled and isolated from the SDN controller.