OpenFlow as a Security Enabling Solution in Large Scale Networked Systems

Gabriella Carrozza, Vittorio Manetti
SESM s.c.a.r.l.
Via Circumvallazione esterna dii
Napoli, Giugliano in Campania, 80014
Naples, Italy
Email: gcarrozza, vmanetti {@sesm.it}

Stefano Avallone, Roberto Canonico, Antonio Marotta
University of Napoli Federico II
Dipartimento di Informatica e Sistemistica
Via Claudio, 21 - 80125 - Napoli, Italy
Email: stavallo, roberto.canonico, antonio.marotta {@unina.it}

I. Introduction

In the Air Traffic Control (ATC) field, where ATC centers belonging to the same system are often deployed over different cities in a given country, and for which pre-operational platforms can be spread over several company premises, Cloud Computing can be considered as a key technology to face several issues. First, setting up an extended enterprise private Cloud Computing platform allows to connect geographically distributed ATC centers for dependability purposes, e.g., by realizing a failover configuration among centers in order to increase overall system availability. Second, it can be leveraged in pre-operational phases to reproduce real world scenarios in house, by setting up testbed platforms in the cloud to perform distributed testing campaigns on complex systems from different premises. However, ATC systems, as mission critical systems, exhibit strict security and safety requirements that rise the challenge of security to be faced at the infrastructure and the application layers, as well as in terms of internal and external vulnerabilities that can threat the overall system. This paper discusses the idea to adopt the layer-2 communication protocol OpenFlow as a security enabling solution in a Cloud Computing environment, and compares OpenFlow based solutions aimed at increasing security in large scale networked systems.

II. OpenFlow and Software Defined Networking

OpenFlow is one of the most promising technologies that guarantees network innovation at a low cost. It was born as a project with the aim of introducing and simply testing new network algorithms and services in the academia context, but its simplicity and its innovating power paved the way to the use of this protocol in a huge number of research fields. OpenFlow can be considered as one of the possible ways for implementing the so-called *Software Defined Network (SDN)* paradigm, which has rapidly changed the perspective of doing network research. The main principle on which SDN is based is a sharp distinction between the infrastructure layer, composed by network devices, and the control layer where the network intelligence is deployed. Using this approach, the forwarding plane is decoupled from the control plane, and network behaviour can be easily programmed through a global view of the network. The OpenFlow protocol is the interface between the two layers and it allows controlling and defining traffic management strategies to be performed by the switch devices in the infrastructure. OpenFlow allows creating new applications on top of existing controllers that can be also tested in simulation/emulation scenarios. The OpenFlow protocol is also attracting interest in cloud computing platforms as a leading technology for implementing networking as a service. The need to have fully-virtualized networks becomes the main focus of the cloud computing community: being the hypervisor the heart of hardware-virtualization, it is needed to find solutions allowing to reach the same level of abstraction with physical network resources. As an example, Nicira company in its Distributed Virtual Network Infrastructure [1] stresses out the need of a fully-virtualized network layer, pointing out the drawbacks of the traditional approaches. The authors underline the need of creating network services that have to be entirely independent from the physical devices at the edge of the network. Another prove of the use of the OpenFlow technology can be found in the recent development of a high number of Quantum plugins [2], that is the OpenStack service that allows to create virtualized networks on a per tenant-basis. Almost all the solutions use OpenFlow to guarantee the programmability of the networking level for the virtual machines. Concerning security requirements, the dynamic nature of cloud computing systems makes the traditional solutions inefficient and that is why there is the need to find new approaches in order to protect such a platform from different kind of attacks. OpenFlow can be considered as a leading technology for implementing an architecture that is aware of dynamic security-policies.

III. A brief survey on OpenFlow based solutions for improving networking security

Some recent works exploit the potentialities of OpenFlow-based platforms to implement security techniques. In [3], authors argue that the SDN paradigm can make the implementation of traffic anomaly detection easier by using the well-known NOX [4] OpenFlow controller in SOHO (small office/home office) networks. They implement four different anomaly detection algorithms as applications on the controller and show that they are able to work at line rate without affecting the home-network related traffic. The results point out the programmability of SDN, in the sense that it allows to implement line-rate detection of network vulnerabilities exploitation and also risk mitigation: once the anomaly is detected, a fast reaction can be spread all over the network. In [5] Khan *et al.* take advantage of an OpenFlow testbed with a NOX controller to implement the detection of P2P traffic based on network layer features. The controller is used to extract network features from the flow-tables of the switches, which are then evaluated and selected using their discriminatory power. This approach can exploit the granularity of the flow-based traffic, instead of the classical deep packet inspection detectors, by guaranteeing a high level of privacy-preservation and accuracy, while having a low deployment and management cost. Braga *et al.* [6] face a known security problem, which is the Distributed Denial of Service. Their proposal aims at minimizing the overhead due to the extraction of network features used in the detection process. They built an application on top of a NOX controller which consists in three different modules:

1) the *Flow Collector* which is in charge of periodically requesting flows to the OpenFlow switches through the secure channel;
2) the *Flow Extractor* which is able to obtain the features involved in the classification of normal-malicious traffic;
3) the *Classifier* that receives the features selected by the module above and uses a self-organizing map (an unsupervised classifier) to detect potential DDoS attacks.

The work shows the great flexibility of implementing a detector by using a standard interface (the OpenFlow protocol) to obtain information about flows in the network. Results underline that the detection and false alarm rates are close to the other approaches, but at the same time it is assured a low overhead rate. This is due to the fact that this approach does not have to collect every packet directed to the victim node to obtain the information needed by the classification process. Wang *et al.* [7] suggest a flexible security management architecture for large-scale production networks to overcome the drawbacks of the existing static solutions, by considering the peculiarities of data center networks. The main aims of such an architecture are:

- service insertion, which means the possibility of introducing new network services without effort (such as protocol identification or a load balancer);
- scalability: to accomplish the need of avoiding a single point of failure, the goal is to assure that introducing more security elements, it is possible to maintain wire-speed performance and to increase reliability of the system.

The proposed architecture is composed by a central control component which has the view of the global state of the network and of the designed security policies; besides one or more security elements which are responsible for detecting anomalies in the traffic patterns. The controller knows which traffic flows (according to the global policies) have to be redirected to a determined security element by simply installing rules in the flow tables of the edge switch. If the security element raises an alarm after the detection process, it can send a message to the controller, which instructs the edge switch to drop all the packets that match the malicious flow. This method represents a powerful way of deciding which security boxes need to be activated in order to implement different security mechanisms or even other services, such as load balancing. However, a distributed architecture requires a deep analysis to verify that the introduced latency (caused by the presence of the security elements) does not affect user experience. The last interesting work about OpenFlow-based security solutions can be found in [8]. A NetServ node can be considered as a programmable node architecture which can allow to design and implement various in-network services and different types of nodes, such as router or switch. It can obviously use OpenFlow switch as data plane: the switch can be connected to NetServ nodes which represent processing units. In the proposed experiment, the authors use a flow-based intrusion detection instead of a deep packet inspection technique. It is a two steps process: the first one, namely flow exporting, consists in creating flows from observed traffic. The second one, namely flow collection, keeps memory of the flows for the subsequent monitoring phase.

IV. Conclusion and future work

The investigation of the proposals found in the literature makes us believe that OpenFlow can be definitively used as a practical solution to increase security and to face vulnerabilities of safety and mission critical applications deployed in a dynamic scenario, namely a Private Enterprise Cloud Computing Infrastructure. As future work, we will define an OpenFlow-based architecture for detecting attacks and implementing mitigation strategies.

References

[1] Nicira *Networking in The Era of Virtualization*. Whitepaper

[2] http://docs.openstack.org/trunk/openstack-network/admin/content/index.html

[3] Mehdi, SyedAkbar and Khalid, Junaid and Khayam, SyedAli, *Revisiting Traffic Anomaly Detection Using Software Defined Networking*. In 2011 Lecture Notes in Computer Science, Recent Advances in Intrusion Detection, pages 161-180 Recent Advances in Intrusion Detection

[4] http://www.noxrepo.org/

[5] H. Khan, S. A. Khayam, M. Rajarajan, L. Golubchik, M.Orr, *Wirespeed, Privacy-Preserving P2P Traffic Detection on Commodity Switches*.

[6] Rodrigo Braga, Edjard Mota, Alexandre Passito, *Lightweight DDoS Flooding Attack Detection Using NOX/OpenFlow*. 2010 IEEE 35th Conference on Local Computer Networks (LCN)

[7] LiveSec: Towards Effective Security Management in Large-scale Production Networks Kai Wang, Yaxuan Qi, Baohua Yang, Yibo Xue, and Jun Li, *LiveSec: Towards Effective Security Management in Large-scale Production Networks*. 2012 32nd International Conference on Distributed Computing Systems Workshops (ICDCSW)

[8] Maccherani E., Femminella M., Lee J.W., Francescangeli R., Janak J., Reali G., Schulzrinne H., *Extending the NetServ autonomic management capabilities using OpenFlow*. 2012 IEEE Network Operations and Management Symposium (NOMS), pages 582-585