**Section 4. GN (Global networks, grid and cloud systems).**

**UDC 004.72**

**Oleksii Cherevatenko, Yurii Kulakov.**

# SECURITY OF PLANES AND LOCAL NETWORKS WITH SDN ARCHITECTURE

The article describes vulnerabilities of SDN-based networks, including local networks and different effective ways to prevent danger to their security and configuration settings. The best practices and methods already tested but not massively implemented are proposed to enable for SDN networks. The applicability of mentioned methods for local networks are analyzed.

**Keywords:** software-defined networks, SDN, security, planes.

Fig.: 2. Tabl.: 0. Bibl.: 7.

**Relevance of the research topic.** As SDN (software-defined networks) became more and more popular in recent years, their security is now considered one of the most important problems in their implementation. SDN most likely has a big future and because of that all questions and controversies of these type of networks need to be answered and resolved, including some single or low-profile solutions for different hardware architectures and topologies.

**Target setting.** Up-to-date, there are several scientific works and articles about SDN security features. However, all of them tell about security options in general and not specified for different solutions. Some information that is required about SDN architectures and topologies, like local SDN topology, is highly required and is a target of this article.

**Actual scientific researches and issues analysis.** Currently, there are some studies about software-defined networks, and their security options and possibilities, but there is lack of researches about local security and architectural interaction in SDN in different levels (planes), so specialists that build SDN topologies do not know about all issues that could possible occur.

**Uninvestigated parts of general matters defining.** Security of SDN and possible issues that could occur during the implementation of their specific solutions and architectures, such as local and single-building topologies, are problems that needs more research. Theoretical basis of these researches can be based on general information about SDN security, with adding specifications of different solutions and topologies and with differentiation to planes.

**The research objective.** The purpose of the article is to determine mechanisms of security of specific SDN solutions on different planes and, primarily, for local

networks and also for a network deployed in a single building but connected to the controller of several networks in different buildings.

**The statement of basic materials.** The architecture of SDN network consists of four levels, or planes, of management which are (also shown in Figure 1) [1]:

- Data plane, which is a set of network devices in SDN topology.
- Control plane, which is the SDN Controller itself. It acts as a link between network devices and applications that provide network configuration.
- Application plane, where different management tools (applications) located. This is the highest level of administration which creates the configuration for the controller and monitors its work.
- Management plane, that connects three other planes so they can interact with each other. It is hidden from network's users [2].

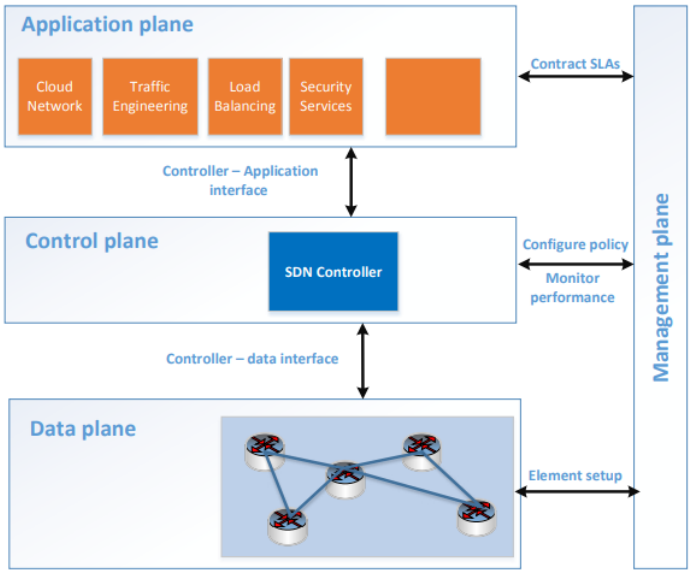


***Fig. 1.*** SDN architectural planes

So, main features that occur interest for implementing security of software-defined networks may be:

- Centralized control of the network and storing information in one place. Because of this security rules can be broadcasted from the controller and can be both general or specific (for example, for a single node in the network), so as the monitoring of the network;
- Applications deployed in the controller are the way how the administrator creates the configuration for the network. Also they are responsible for monitoring and updating the network, load balancing, storing data, maintaining connections.

The separation of data and management in software-defined networks means that new approaches can be used to approve the security. They are different from

standard IP networks because of the high-level automation used in this type of technology and specific interaction between planes of administration.

Let's figure out how three main planes (excluding the management plane that is intended to connect the others and cannot be possibly vulnerable) can be affected with different types of insecure actions (Figure 2) [3].

In Data Plane, the main threat is flow table flooding which causes the overload of flow tables which are responsible for routing in SDN. More units in the network means larger flow tables. If network has many active flows which interact all the time, flow tables can be easily overloaded by adding fake or wrong routes. Obviously, this will cause blocking of adding new lines into tables and the network will completely fail soon. The connection between the control plane and data plane when someone but the controller has access to it (which means human) called "Man-in-the-middle". The way to prevent this is to decrease SDN Controller workflow so it can deliver flow tables more effectively and also reduce packet loss by upgrading the quality of connections (for example, enable better transport/network protocol). To do this, the administrator may need a specific solution. One that has proven its effectivity is STAR, tested in Spanish backbone network and demonstrated decreasing workflow of the controller by 87% and reducing packet delay by half [4].

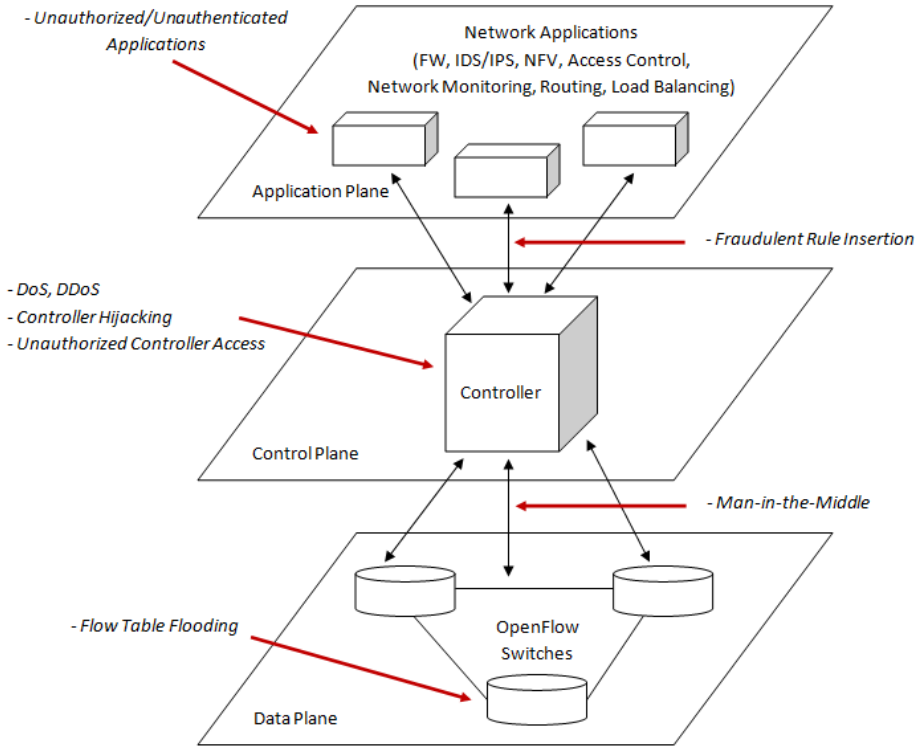


***Fig. 2***. Secure threats to generic SDN-based architecture

Control Plane has several vulnerabilities. Management in SDN networks based on external services (which the controller may host as applications). From Figure 1, it is seen that applications interact with the controller via application programming interface (API). Because protocols, that usually in use for SDN management, allow all type of traffic, they can possibly allow dangerous traffic. The controller does not distinguish safe and dangerous packets until it accomplishes the investigation when it will be too late. Because of that the controller can be a target for DoS or DDoS attacks.

To prevent that, some standard protection technologies such as routing policies, individual link protection, OSPF areas can be used. For example, OSPF areas can make the network units (or group of them) independent from the other network and some parts of the topology can stand the attack. In local networks it may be critical, because using individual protections and areas can prevent the attack from the outside with big probability. Using own tables and even protocols for local network is a best practice for security.

Another threat on this plane is the possibility of unauthorized access to the controller. It can be physically damaged or hacked but the only way to stop that from happening is to protect the facility where the controller physically located, use password protection and encryption. These things are not directly related to network technologies and are not the topic of this article. But the actual problem is the remote unauthorized access to the controller which can be done with using network connections and malware. After accessing the controller intruder can break down the network by adding wrong rules and routes, or run a virus in the system which can do program and physical damage.

One of the ways to prevent access like this is to deploy a special application on the controller which will regulate the access to the controller by filtering IPs, user logins and permissions. For example, permissions can be separated to read configuration, add new rules and applications, update them and delete them. Depending on the needed security of the network, different users can access the controller with certain permissions – for example, the main administrator has all permissions and can do everything, software engineer can update applications due to updated requirements, technical stuff can only read the configuration, etc. All of this, including the regulation of permissions, can be done automatically by software, like SDN Controller Dynamic Access Control System which can prevent API abuse of the controller (malicious traffic from the applications). It can work independently from the Controller and prevent damage to it with low performance load (only 0,5% overload) [5].

The application plane contains vulnerabilities which may be caused by unauthorized applications that can do damage to the whole network or some certain units in it. Because of plenty of possible applications, many aspects of the SDN can be damaged, so it requires quality security approaches.

As it was told earlier, password and encryption security are crucial. The more time malware application will try to access the controller, the bigger possibility that the intruder would be detected. So if the controller is a PC with special hardware and settings, it is necessary to have:

- Most recent updates of the software, which includes security updates. For example, latest (up to date of publication of this article) Microsoft Windows 10 and Server 2019 versions often receive updates with detailed description.
- Most recent updates of hardware drivers, which every manufacturer has.
- Two or three step verification of user that needs access to the controller and its applications. Best practice is to develop extra application so it will run only on one specific controller (no need to use third-party software) and regulate permissions that were described earlier.
- Use encryption of data when planes of SDN architecture interact with each other. This will complicate recognition of network's configuration for third-party users, who can possibly be malefactors [6].

In this article, we are interested in local SDN network security so the technology of observing set of flows by the controller may be useful in here. The technology includes the method of checking frequent sets to automatically suspect possible damage. SDN network based on flows that are responsible for exchanging data. Every flow in this solution represented as a set. In turn, the set consists of different data that is needed for units to connect such as network protocols, addresses, ports, number of packets, etc. Each set also contains information about network it comes from. The controller has an option called MinimalSupport that can be set manually and responsible for the number of data sets (flows) that the controller observes. For local networks data sets would be headed as <tcp, 192.168.1.X, *>, where "tcp" is the transport protocol, "192.168.1.X" refers to the network and "*" is a place for ports. If the controller finds network that are not frequently used to access the controller, it can block the traffic from there with the special application. The administrator will be warned and can do further activities to prevent danger to the network. This method can prevent unauthorized traffic from the outside and protect controller apps which are the key to make configuration of the network [7].

**Conclusions.** This article describes what type of vulnerabilities each plane of SDN network has and how to protect units and applications of the network in effective ways. It is shown in the article that SDN network security should be implemented separately on each level, still the overload of the network will not exceed the standards.

## References

1. Cabaj, K., Wytrębowicz, J., Kuklinski, S., Radziszewski, P., Dinh, K. (2014). SDN Architecture Impact on Network Security. In Proceedings of the Federated Conference on Computer Science and Information Systems, Warsaw, Poland, Volume 3 (pp. 143–148).

2. Kulakov, Y., Kohan A., Kopychko S. (2019). Traffic Orchestration in Data Center Network Based on Software-Defined Networking Technology. In Proceedings of the 2nd International Conference on Computer Science, Engineering and Education Applications (ICCSEEA2019) (pp. 228-237).

3. Akbaş, M., Karaarslan E., Güngör C. (2016). A Preliminary Survey on the Security of Software-Defined Networks. In Proceedings of International Conference on Advanced Technology & Sciences (ICAT'16), Konya, Turkey (pp. 468-473).

4. Guo, Z., Liub, R., Xuc, Y., Gushchind A., Walide, A., Chao, J. STAR: Preventing flow-table overflow in software-defined networks. Computer Networks, 2016, vol. 125, pp. 15-25.

5. Tseng, Y., Pattaranantakul, M., He, R., Zhang, Z., Naït-Abdesselam, F. Controller DAC: Securing SDN controller with dynamic access control (2017). In Proceedings of IEEE International Conference on Communications (ICC), Paris (pp. 1-6).

6. Hogg, S. (2014). SDN Security Attack Vectors and SDN Hardening. [Online] Retrieved from: https://www.networkworld.com/article/2840273/sdn-security-attack-vectors-and-sdn-hardening.html.

7. Agrawal, R., Imielinski, T., Swami, A. (1993). Mining Association Rules Between Sets of Items in Large Databases. In Proceedings of ACM SIGMOD Int. Conf. Management of Data. [Online] Retrieved from: https://dl.acm.org/doi/10.1145/170036.170072.

## AUTHORS

**Oleksii Cherevatenko** – student, Department of Computer Engineering, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute".

E-mail: chereva@ukr.net

**Yurii Kulakov** (supervisor) – professor, Doctor of Engineering Sciences, Department of Computer Engineering, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute".