# SDN ARCHITECTURE FUNDAMENTALS AND DOS PREVENTION BASICS: A CASE STUDY WITH OPENFLOW

**Gökhan AKIN**
*Trakya University*

**Enis Karaarslan**
*Mugla S.K. University*

**Ozan BÜK**
*İTÜ*

**Erdem UÇAR**
*Trakya University*

**Abstract**

*The design of today's network switching solutions goes back to the architecture designed in the 1970s. There is no significant change that took place despite IPv6 protocol. However, access speeds and the number of users are increasing continuously. There is a need for better network management and security. The Openflow protocol and Software Defined Network (SDN) architecture is a revolutionary change in network management and monitoring operations. In this study fundamentals of SDN architecture is given and a basic DoS detection and prevention system that can be performed with SDN is demonstrated.*

**Keywords:** SDN, Software Defined Network, Openflow, Ethane, Denial of Service, DoS, Mininet

## 1. INTRODUCTION

Today, more than 3 billion users are connected to the Internet via a total of 51171 Autonomous Systems[1]. Both the number of users as well the speeds of Internet access are continuously increasing. The number of devices that must be kept under control and the increase of network traffic results in network management problems. SNMP and some other protocols were used to manage network devices and servers. Although these methods simplify monitoring and management, it cannot be a solution to the increasing number of devices and security threats. TCP/IP protocol architecture has been the same for more than 40 years, except the development of IPv6 protocol. Security vulnerabilities in the design of Internet Protocol and the lack of security awareness caused an increase in the number of malicious activities seen on the Internet.

There is a need to find a way by which we can see and control the traffic flows in the network.

In this paper, SDN based LAN management and DoS prevention techniques are demonstrated which is one step further to solve current network problems. In the first section the fundamentals about network architectures and security problems will be explained. Then the implementation and experimental results will be given. Lastly possible future work will be discussed.

## 2. FUNDAMENTALS

In this section, network architectures and security considerations are discussed.

### 2.1 CURRENT NETWORK DEVICE ARCHITECTURE

In the development process of network technologies, first routing decisions are carried out on a Honeywell DDP-516 minicomputer which has a specially developed software[2]. The packet switching processes which has started with static routes, continued with dynamic routing protocols which were added over time. These devices switch packets with a central processor according to routing tables that they have created statically or dynamically.

With the increasing Internet access speeds, it became more difficult to meet the needs of

packet switching with software solutions which requires the use of expensive processors. This also increased energy consumption and cooling needs. As a solution, hardware-based architectures were developed which has high packet switching capabilities. Hardware-based architectures are able to provide high packet switching capacity for a much cheaper cost[3]. Adding features that arise over time to these hardware is only possible by the design and production of new hardware. This is a long and costly process. For this reason, today's routing devices architecture is composed of control plane and data/forwarding plane.

Control plane, handles the processes such as routing protocols, ARP mechanism, etc. which are managed by software. Data/forwarding plane enables hardware packet switching to the chosen destination interface after the control plane process.

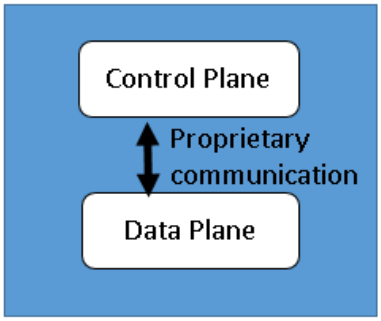


***Fig. 1.*** *Today's routing devices' architecture*

As it can be seen in Figure 1, Control Plane and the Data/Forwarding Plane are configured in the same hardware and the communication between them is carried out by proprietary protocols developed by the manufacturing companies. In this structure every router and switch makes its own final decision. There is no central decision mechanism. The amount of network devices and network traffic is increasing and there is a need for central management.

## 2.2 CENTRALIZATION EFFORTS IN NETWORKING

In 2004, some studies have been performed such as Route Control Platform (RCP) but remained as a special study for BGP (Border Gateway Protocol)[4]. In 2007, a study conducted at Stanford University revealed a new architecture named Ethane[5]. "Ethane is a new architecture for enterprise networks which provides a powerful yet simple management model and strong security Guarantees."[6]. The architecture that was described as, later on paved the way for SDN and Openflow protocols.

SDN (Software Defined Networking) is an architecture that enables to move the Control Plane to another machine (control server) in a way that enables researchers to interfere the processes, do research, develop software, and create new technology. But in this case, a protocol is needed to provide communication between Control Server and the Data/Forwarding Plane. For this purpose, the first protocol developed and widely used is Openflow protocol[7].

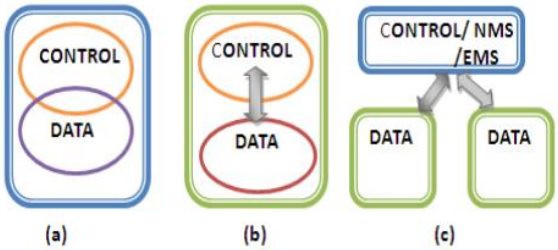


***Fig. 2.*** *Different architectures in today's networks [7]*

Today's network devices' architecture evolved from central CPU based architecture (Fig. 2.a), to a layered architecture (Control Plane and Data/Forwarding Plane) (Fig. 2.b). Now SDN architecture is evolving (Fig. 2.c).

Openflow protocol[7] has provided an open architecture for the communication between the Control Plane and Data/Forwarding Plane. It became possible to develop the network control software that can be installed on a server and work independent from the hardware. It has become possible to do research and develop new features on the Control Plane.

## 2.3 SDN ARCHITECTURE

SDN architecture can be divided into three parts of its structure as shown in Figure 3. The lower part consists of network devices. These devices are connected to the control server in the central part via southbound interface. In the upper part there are APIs (Application Programming Interface) that connects to the

control server via northbound interface. In this structure, all the control functions are shifted to the control server, so the network devices at the bottom became data layers that only handles packet exchange. The communication between the control layer and data layer interfaces are provided with the northern interface APIs. Openflow protocol is a running example that provides this architecture. This structure allows to manage all network devices from a single point.

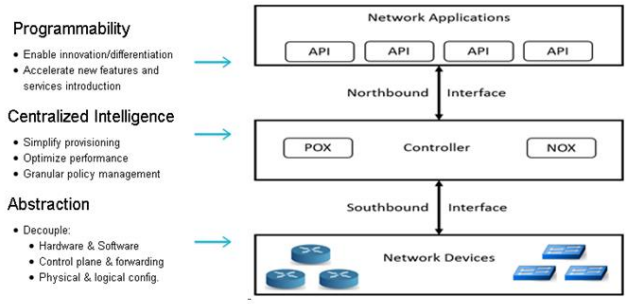


***Fig. 3.*** *SDN Architecture*

## 2.4 SECURITY SOLUTIONS AND WEAKNESSES

There are various network security solutions available, such as firewalls, Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS).

The main purpose of a firewall is to protect the internal network from external networks. Firewalls consider the internal network safe as a working principle. However, although it may not seem as a threat for the firewalls, some packets like DNS queries, SYN packets can be a threat vector. IDS/IPS have been developed with the aim to identify such initiatives. These devices monitor internal and external network traffic in both directions, generate alarm about the various attacks and abnormal activities observed in the traffic, generate logs and block traffic if desired.

IPS try to identify the attacks by using signature-based and flow-based traffic analysis. However, they can lead to problems with increased traffic volumes. Problems encountered in the management of the IPS system can be divided into three parts:

- Latency: The inline positioning structure of the IPS system in the network results the inspection of each packet passing through the IPS. This will reduce network performance as well as lead to delays. Delay sensitive systems will be affected negatively so IPS systems may have to be kept in a passive mode for such traffic.
- Accuracy of decisions: The accuracy of the alarms generated by IPS systems are still not optimized to the desired values. False-positive detection of innocent traffic remains one of the biggest problems encountered in IPS management[8]. Management of the IPS systems requires advanced expertise.
- Positioning: The IPS system is positioned on the entrance and exit of the network, which causes the internal network traffic to become uncontrolled. The analysis of internal network traffic within the IPS is not considered as a practical solution that will bring huge burden in terms of cost[9].

## 2.5 SDN BASED SECURITY SOLUTIONS

One of the features that come with SDN is increasing the traceability of the flow of traffic through the router and switch devices. Shifting the network anomaly detection capability toward the source further increases the effectiveness of this kind of systems.

The use of flow-based systems to detect DDoS attacks are mentioned in previous studies [10-11]. Anomalies observed in the network traffic environment can be categorized in three parts [12]:

- Network operation anomalies: Network operations are such as changes made by the network management; increasing the bandwidth, introduction of a new QoS profile and etc.
- Flash crowd anomalies: The traffic burst can be compared as a rush for exam results announced, or the rush after an announcement of a critical software update. These are expected situations.
- Network abuse anomalies: The abuse of the network anomalies are observed during actions such as Network DoS/DDoS or port scan. The amount

of traffic anomalies of this type may not always result in high volume of traffic. However, a significant increase is observed in the number of packets and the number of connections for this anomaly based flows[13].

Openflow supported network switches provide query capability of the data flow tables which increases the visibility of the traffic on the network. This can be used for the construction of anomaly analysis and prevention of undesired network traffic. In traditional networks, this process could only be done with reactive methods in remote locations such as backbone or distribution level switches that are far from the source. This process can be moved to the access layer with SDN that can proactively drop the traffic at a level that is closer to the source of traffic.

Figure 4 shows the graph of the attack that took place in the main router of the Istanbul Technical University campus network on December 15, 2014. The average number of connections per second (which was observed in 100,000 level) is observed as 833,000 level during the attack.

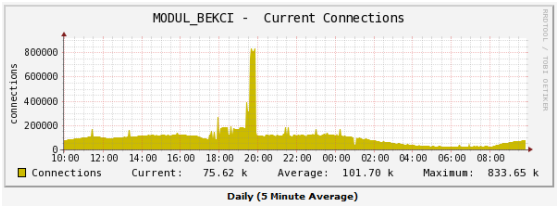


**Fig. 4.** Sample DNS Attack[13]

As it can be seen in Figure 4, it can be easily observed to what extent these attacks have effect on real time traffic. The solution can be possible by using programmable structure of SDN architecture and using the flow data that is available in an Openflow supported switch device.

## 3. EXPERIMENTAL WORK

Experimental work is implemented in two phases. The aim is to see the benefits of SDN architecture in a LAN environment and implement a basic DoS prevention system. In this study, a demonstration was implemented as a solution to DNS DoS attacks using the Mininet platform and Python language solutions through APIs. A performance study[14] was conducted as a joint project to determine the benefits of SDN architecture on the campus network system. In this study, Openflow Protocol is used to make measurements with open source D-ITG[15] software, using Openflow enabled HP network switch (HP E5406 zl). In this study, bandwidth and delay measurements were made in three different scenarios. (1)Openflow protocol is off, (2)Openflow protocol is on and (3)external controller is activated. The performance impacts of using an external controller was examined accordingly. In this study, a network data traffic is generated in the test environment with five different sizes (64, 128, 512, 1024 Byte) each for 8 hours. Bitrate values are shown in Figure 5a. In Figure 5b, Round Trip Time delay is shown which is measured with D-ITG application. No significant difference was observed when an external Openflow controller is used.

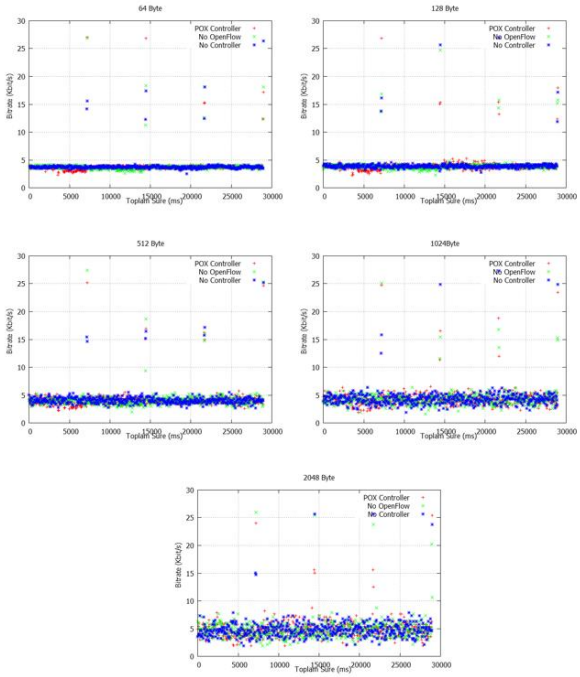


**Fig. 5a.** Bitrate Values [12]

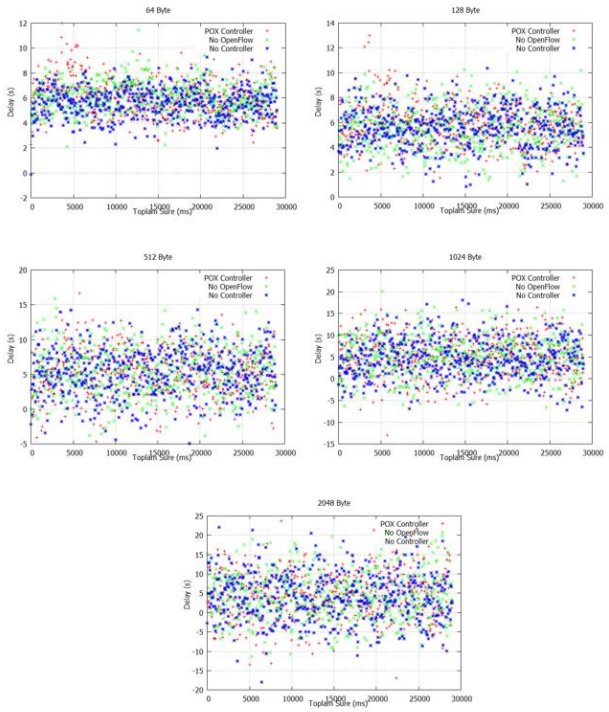


**Fig. 5b.** Delay Values [12]

In this study, we will primarily focus on the advantages of Openflow Protocol that provides control from a single center.

The programs such as QoS, firewall, intrusion detection and prevention, routing etc. can communicate with the control server via northbound interface APIs. Intrusion detection can be made with packet statistics received by Openflow switch through Northbound interface. After the detection of intrusion, a study was made to stop the source of the attack. During the study, a network topology is established using Mininet that contains two hosts, an Openflow switch, and a control server. POX is used as a control server, Open vSwitch is used as a switch. As the working environment Mininet virtualization package is built on Microsoft's Windows 8 operating system running on Virtual Box virtualization platform. The control server POX has been selected that comes built-in with Mininet. A Python code that was created on POX, queries the port and flow information from the Openflow switch device every 5 seconds. Using the data obtained from this queries, it has been targeted to drop DNS traffic when the number DNS packets entered from a client exceeds a certain threshold value.

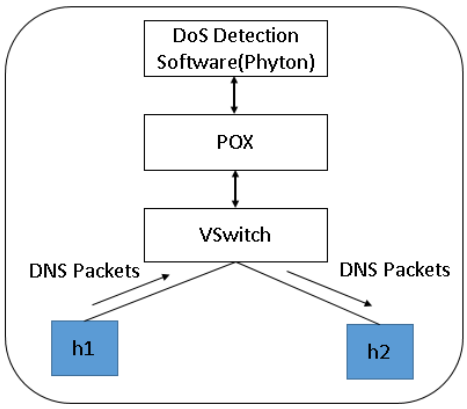


**Fig. 6.** Mininet Test Pad

The determination of the threshold value is not the main purpose of this study. However, to determine a realistic value a test environment has been established in ITU with 7 clients Internet traffic was monitored for 6 hours. The average time interval of the number of DNS packets observed during this period was found 6.3 in 5 seconds time intervals.

$$avg = \frac{\sum_{user=1}^{7} \frac{\Delta packet_{user}}{7}}{\Delta t} \quad (1)$$

The average threshold value to be used in order to be a bit more flexible for DNS anomaly detection is set as 10.

Iperf packet generator is used to send fake DNS packets. To do this, on the second host, iperf server has been launched that uses UDP port 53 that is used in DNS.

$ iperf -u -s -p 53

In order to send bogus DNS packets from the first host to the second host, the following command is entered in the first host interface.

$ iperf -u -c 10.0.0.2 –n 10 –t 5

This command sends 10 UDP packets in every 5 seconds to H2 that uses the IP address 10.0.0.2. This configuration forms the basic test environment. In the second phase of the study the number of DNS packets sent from H1 is increased. If traffic exceeds the threshold it will decide that the traffic is abnormal, and will send a command to the Openflow switch with a code programmed in the bash, to drop the matching DNS traffic packets in the flow table. Thus the harmful traffic is dropped before its inclusion to the network.

In the second stage iperf fake DNS traffic value was set to 20. The graphs of logs obtained after the trials showed that traffic is being dropped in port after the attack.

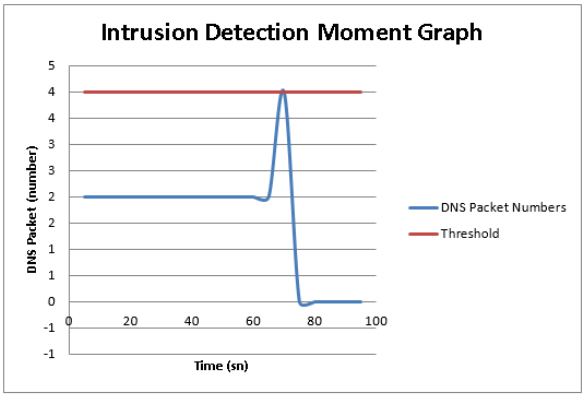


**Fig. 7.** Intrusion Detection Moment

## CONCLUSION

In this study, Software Defined Networking architecture, a new paradigm in the computer networks world has been examined. Performance tests which were implemented in a previous joint study are examined to detect the benefits of SDN in a LAN environment. After observing minor changes in terms of performance values, central control advantages of SDN has been focused. SDN based simple DoS detection system has been created which uses the flow information received from each network device. In this way, anomaly detection can be performed within the LAN on each switch. Observed anomalies can be used to drop this kind of traffic. SDN based systems could be used to prevent DoS attacks effecting any part of the internal network which cannot be done with the traditional IDS / IPS systems.

This study demonstrated the detection and prevention of DNS-based attacks. In further studies other attack signatures that may occur on the network can be identified and how to drop these traffic with Openflow Controller can be determined. A possible solution is directing only the traffic exceeding certain thresholds to an IDS system for even more detailed analysis [16]. This can ensure to analyze only the traffic that could pose a risk instead of analyzing all the traffic.

## REFERENCES


[1] The Cooperative Association for Internet Data Analysis. "AS Ranking" http://as-rank.caida.org/, Last access: October 10th, 2015.

[2] Router, Historical and technical information, Wikipedia Encyclopedia, https://en.wikipedia.org/wiki/Router_(computing), Last access: October 10th, 2015.

[3] Casado, M., Koponen, T., Moon, D., & Shenker, S., Rethinking packet forwarding hardware. In Proc. Seventh ACM SIGCOMM HotNets Workshop (pp. 1–6), 2008.

[4] Shaikh A., Route Control Platform Making an AS look and act like one router, AT&T Labs – Research, IEEE CCW 2004, 2004.

[5]Martin Casado, Michael J. Freedman, Justin Pettit, Jianying Luo, Nick McKeown, Scott Shenker. "Ethane: Taking Control of the Enterprise," ACM SIGCOMM '07, 2007.

[6] Ethane: A Security Management Architecture Official Web Site, Stanford University, http://yuba.stanford.edu/ethane/, Last access: October 10th, 2015.

[7] Das S. , Parulkar G., McKeown N., Unifying Packet and Circuit Switched Networks, Department of Electrical Engineering, Stanford University, 2009.

[8] Khayam S.A., Khalid J., Mehdi S.A., 2011, Revisiting Traffic Anomaly Detection Using Software Defined Networking, R. Sommer, D. Balzarotti and G. Maier (Eds.): RAID 2011, LNCS 6961, Springer-Verlag Berlin Heidelberg, 2011.

[9] Zhang L., Shou G., Deployment of Intrusion Prevention System Based on Software Defined Networking, IEEE 978-1-4799-0077-0/13, 2013.

[10] N. Ye, "A markov chain model of temporal behavior for anomaly detection," in Workshop on Information Assurance and Security, West Point, NY, 2000.

[11] D. Moore, G. Voelker, and S. Savage, "Inferring internet denial of service activity," in Proceedings of 2001 USENIX Security Symposium, Washington, DC, 2001.

[12] Barford P, Plonka D., Characteristics of network traffic flow anomalies, IMW'01, November 1-2, San Francisco CA, USA, 2001

[13] Intrusion and Prevention in SDN Network, Uysal M., İTÜ, 2014.

[14] Yazar S. Openflow Technology, Network Switches which are Openflow enabled and Investigation POX Controller, Msc. Thesis, Trakya University, 2013.

[15] D-ITG, Distributed Internet Traffic Generator Official Wep Page, http://traffic.comics.unina.it/software/ITG/, Last access: October 10th, 2015.

[16] Xing T, Huang D, SnortFlow: A Openflow-based Intrusion Prevention System in Cloud Environment, IEEE DOI 10.1109, 2013