ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 5, Issue 12, December 2017

Leveraging SDN and WEBRTC for Rogue Access Point Security

Prof. Kanchan Varpe, Komal Khandale

RMD Sinhgad School of Engineering, Pune, India

Student, RMD Sinhgad School of Engineering, Pune, India

ABSTRACT: Rogue access points (RAPs) are unauthorized devices connected to a network, providing unauthorized wireless access to one or more clients. Such devices pose significant risk to organizations, since they provide a convenient means for hackers and insiders to hide malicious or unsanctioned activities on industry, government, and campus networks. Yet, limitations inherent in traditional networks make detecting and removing such devices expensive, time consuming, and difficult to implement. For software-defined networks (SDNs), the risk of a network compromise due to RAPs is equally concerning, and methods for detecting RAPs within SDN architectures are needed. Hence, our work leverages the capabilities of an SDN along with a Trusted Agent (TA) to detect and deny RAPs access to networks by using both generic and novel methods with minimal impact to performance. Three other contributions are included in this work. They include: 1) utilizing an emerging web architecture (webRTC) to detect hidden subnets; 2) developing the first, security-based, use case for Mininet-WiFi, a software defined wireless network (SDWN) emulator; and 3) enhancing Ryuretic, a modular programming language for SDN application development.

KEYWORDS: Network Address Translation (NAT); Rogue Access Point (RAP); Malicious Access Point; Software-Defined Networks (SDN); Ryuretic; Ryu; Network Security; Intrusion Detection Prevention System (IDPS); Trusted Agent; WebRTC

I. INTRODUCTION

1.1 Software Defined networking

In contrast to traditional networks, a Software-Defined Network (SDN) separates the control and data planes, so network intelligence and states are nearly abstracted away and maintained by a logically centralized controller. Simply say, SDN decouples the control logic from vendor-specific hardware. Since SDN allows network operators to implement network applications at a higher level of abstraction.

1.2 Network Security

Network security involves the authorization of access to data in a network, which is controlled by the network administrator. Network security consists of the policies and practices adopted to prevent and monitor unauthorized access, misuse, modification, or denial of a computer network and network-accessible resources.Network security is a growing concern for government, industry, and campus networks, and hackers continuously test the limits of the security tools available to network operators

1.3 Rough Access Points

Rogue Access Points (RAPs), sometimes called malicious APs, such as wireless switches (WS) and wireless routers (WR), are no exceptions. Generally speaking, RAPs are unauthorized wireless devices that, when connected to an organization's network, provide unauthorized wireless connectivity to other clients.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 5, Issue 12, December 2017

1.4 Network Address Translation

Network address translation (NAT) is a method of remapping one IP address space into another by modifying network address information in Internet Protocol (IP) datagram packet headers while they are in transit across a traffic routing device.

Beyond NFG's focus on SDN-based, RAP detection and isolation, our work also offers the following contributions:

- Establishes an initial framework for detecting and blocking rogue devices connected to SDN infrastructures
- Provides a public, open-source implementation of our work on GitHub [8]
- Provides a first use case for security emulation with Mininet-WiFi [9]
- Utilizes a combination of passive and active detection techniques to lessen the SDN controller's computation
- Uses the webRTC architecture to provide a novel NAT/RAP detection measure
- Improves features of Ryuretic [10], a modular programming framework for SDN application development

II. LITERATURE SURVEY

2.1 A novel approach for elimination of RAP in wireless network

In this paper, we have discussed existing techniques for detection of rogue AP with its effectiveness and weaknesses. And also describe our solution for this major concern in wireless network. Now a day's wireless LAN is widely used in many public spaces. Wireless access points expand wired network. It provides more flexibility to the users. There is a big risk that users connect to rogue access point (Rogue AP). Detection of rogue AP is a challenge for network administrator. Undetected rogue APs are serious threats which steal sensitive information from the network. There are many techniques used for detection of fake AP .

2.2 Detecting rogue access points using client-side bottleneck bandwidth analysis

This paper aims at surveying different methods of rogue Access Point detection. Rogues are unwanted whether they are access points or clients as they steal critical data and bandwidth. To detect a rogue access point different approaches are used. These approaches are briefly classified as Client-side approach, Server-side and Hybrid approach. Every approach has its own advantages and disadvantages. Clients have limited resources and do not possess much control over network when compared with servers. Amongst all approaches Hybrid approach is efficient because it minimizes the inabilities of client side approach and adds server control for detection of Rogue Access Points (AP).

2.3 Mininet-wifi: Emulating software-defined wireless networks

This paper introduces Mininet-WiFi as a tool to emulate wireless OpenFlow/SDN scenarios allowing high-fidelity experiments that replicate real networking environments. Mininet-WiFi augments the well-known Mininet emulator with virtual wireless stations and access points while keeping the original SDN capabilities and the lightweight virtualization software architecture. We elaborate on the potential applications of Mininet-Wifi and discuss the benefits and current limitations.

2.4 Passive NAT detection using http logs

Network devices performing Network Address Translation (NAT) overcome the problem of the deficit of IPv4 addresses as well as introduce a vulnerability to the network with possibly insecure configurations. Therefore detection of unauthorized NAT devices is an important task in the network security domain.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 5, Issue 12, December 2017

2.5 A timing-based scheme for rogue AP detection

This paper considers a category of rogue access points (APs) that pretend to be legitimate APs to lure users to connect to them. We propose a practical timing-based technique that allows the user to avoid connecting to rogue APs. Our detection scheme is a client-centric approach that employs the round trip time between the user and the DNS server to independently determine whether an AP is a rogue AP without assistance from the WLAN operator

2.6 Enhancing network security through software defined networking (SDN)

Software Defined Networking (SDN) is an emerging technology that attracts significant attention from both industry and academia recently. By decoupling the control logic from the closed and proprietary implementations of traditional network devices. SDN can help enhance network security and information security process

2.7 A passive approach to rogue access point detection

Unauthorized or rogue access points (APs) produce security vulnerabilities in enterprise/campus networks by circumventing inherent security mechanisms. We propose to use the round trip time (RTT) of network traffic to distinguish between wired and wireless nodes. This information coupled with a standard wireless AP authorization policy allows the differentiation (at a central location) between wired nodes, authorized APs, and rogue APs. We show that the lower capacity and the higher variability in a wireless network can be used to effectively distinguish between wired and wireless nodes. Further, this detection is not dependant upon the wireless technology (802.11a, 802.11b, or 802.11g), is scalable, does not contain the inefficiencies of current solutions, remains valid as the capacity of wired and wireless links increase, and is independent of the signal range of the rogue APs.

III. MOTIVATION

Computer networks are large and complex, many consisting of over 1,000 network devices (e.g., routers, switches, firewalls, IDPS, and multiple other middleboxes). In spite of its complexity and size, network configuration still relies largely on manual configurations. Even within SDN environments, network operators may find themselves repeatedly interacting with SDN controllers in order to reset the policy enforcements that are activated by security policies. As a result, they need abstractions to simplify the development of network applications and to enforce policies. Likewise, network operators need frameworks that reduce their involvement with network policy enforcements and network configurationchanges. And, where possible, they can benefit from an ability to pick and choose which network modules they wish to include in their SDN. we first demonstrate the capabilities of a modular programming framework to couple applications together to enforce multiple network policies and improve network security.

IV. PROPOSED SYSTEM

The proposed system carries the task in four phases as shown in fig. Throughout the phases, the work transitions from simple detect and deny security applications to a much more robust system for network security and management.
In the first phase, the work investigates the capabilities of SDN to mitigate existing attack vectors (e.g., rogue DHCP server detection [33] and spoofing of ARP packets [34]) along network edge-devices. We find that these solutions are easily implementable because SDN makes a tremendous amount of state information available to edge-devices. Consequently, this availability of state allows for the rapid development of new, extensible, and inexpensive SDN-based solutions that augment a mac-learning switch to mitigate edge-based network attacks. Moreover, these SDN-based solutions are easily adaptable to current network infrastructures (and protocols) or cloud environments.
The second phase, recognizes certain limitations of the framework used in the first phase. As a result, this phase introduces a programming framework, introduced as Ryuretic [5], for modular, fine-grained, SDN application development. Accordingly,network operators are able to utilize Ryuretic's abstractions to implement more robust network security and management applications for their networks.
In phase three, the weakness of the previous phases, which required the network operator's continued involvement with network policy configurations, is addressed. It offers a Trusted Agent [30] to help automate simple security policy

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 5, Issue 12, December 2017

transitions and further capitalize on the promise of SDN and NFV to reduce network operator workloads. Moreover, by introducing a Trusted Agent, automated network configurations serve to eliminate manual configuration errors relating to policy enforcement transitions

Finally, the fourth phase, builds off the previous phases to introduce expanded network management capabilities, and it introduces an active measure for detecting malicious activity. These features are made possible by the Ryuretic programming framework and the adoption of a Trusted Agent.

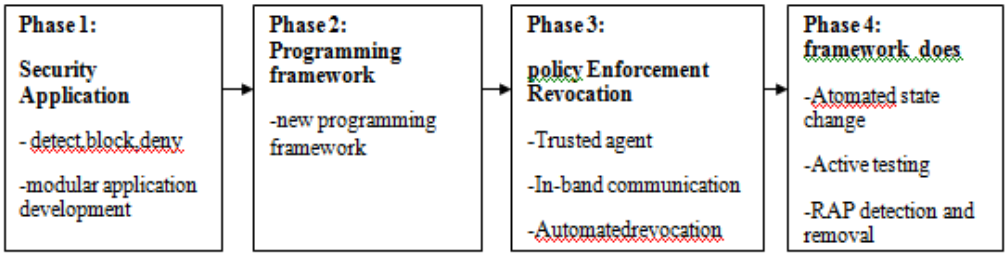

Fig1.Proposed system function

Phase 1.1 Leveraging SDN to Improve the Security of DHCP:

Current State of the art technologies for detecting and neutralizing rogue DHCP servers are tediously complex and prone to error. Network operators can spend hours (even days) before realizing that a rogue server is affecting their network.
We present Network Flow Guard (NFG), a simple security application that utilizes the software-defined networking (SDN) paradigm of programmable networks to detect and disable rogue servers before they are able to affect network clients
Phase 1.2 Leveraging SDN for ARP Security:
Insider threats are a growing concern for industry, government, and campus networks. software-defined networking (SDN) enables the implementation of novel security measures that are capable of detecting and eliminating ARP (Address Resolution Protocol) spoofing before it can impact other hosts. Hence, this phase presents Network Flow Guard for ARP (NFGA), an SDN security module that augments simple, MAC-learning, protocols on OpenFlowenabled0 switches. NFG works by hashing a host's physical address with an appropriate port-IP association to deny ARP spoofing at real-time.

Phase 2 : Ryuretic: A Modular Programming Framework for Ryu

We present Ryuretic as a modular, programming framework for SDN application development. Ryuretic draws its inspiration from Pyretic, which already offers powerful, modular abstractions for network operators; however, Ryuretic also builds on Ryu's greater variety of packet match fields and its access to advanced OpenFlow protocols. This framework allows programmers to create new, extensible, and more powerful network applications at a much higher level of abstraction.

Phase 3: Security Policy Transition Framework for Software Defined Networks

Controllers for software-defined networks (SDNs) are quickly maturing to offer network operators more intuitive programming frameworks and greater abstractions for network application development. Likewise, many security solutions now exist within SDN environments for detecting and blocking clients who violate network policies. However, many of these solutions stop at triggering the security measure and give little thought to amending it. Hence, we present a security policy transition framework for revoking security measures in an SDN environment once said measures are activated.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 5, Issue 12, December 2017

Phase 4: Leveraging SDN and WebRTC for Rogue Access Point Security

Rogue access points (RAPs) are unauthorized devices connected to a network, providing unauthorized wireless access to one or more clients. Such devices pose
risk to organizations, since they provide a convenient means for hackers and insiders to hide malicious or unsanctioned activities. For software-defined networks
(SDNs), the risk of a network compromise due to RAPs is equally concerning, and methods for detecting RAPs within SDN architectures are needed. Hence, our work leverages the capabilities of an SDN along with a Trusted Agent (TA) to detect and deny RAPs access to networks by using both generic and novel methods with minimal impact to performance.

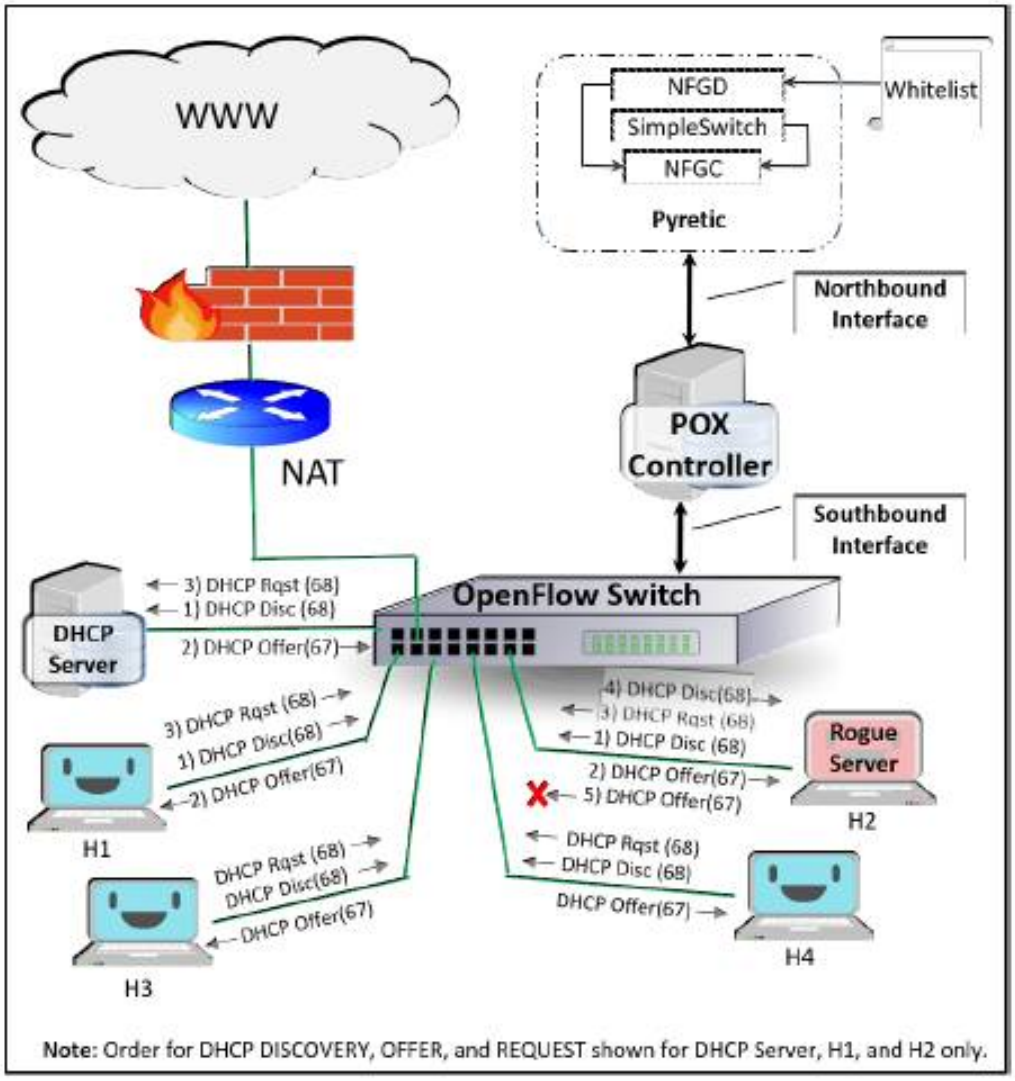

Fig. NFG Implementation.

The Network Flow Guard (NFG) design enables existing network protocols so as to minimize its impact to the current network topologyNFG capitalizes upon Internet standards specified in RFC 2131 [43] requiring DHCP to use UDP as its transport protocol and the Bootstrap Protocol (BOOTP) [44] requiring specific source and destination ports be used.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 5, Issue 12, December 2017

As depicted in Figure, we utilize the Pyretic [16] framework to develop NFG's Coupler (NFGC) and DHCP (NFGD) modules. The NFGC module couples our NFGD module with Pyretic's native MAC-learning (or simple) switch applicationThis coupling allows the NFGD module to run concurrently with the switch application and is responsible for filtering suspicious network traffic before it can affect DHCP services on the networkWhen validating DHCPOFFERS, NFGD references a preloaded whitelist (maintained by the network operator) of approved DHCP servers. This list can also be updated at Since networks often contain a combination of both dynamic and static addresses, NFGsupports both static and dynamic port allocations simultaneously. This is accomplished by allowing network operators to submit static MAC-IP-port allocations via a static-list file as indicated. This list is then added to NFG's dynamic table at boot and protected from overwriting via a fixed field permission check. The remaining entries are then added via the Dynamic Table Updater module, and the table is used by the ARP Validator to determine the validity of ARP-reply packets.

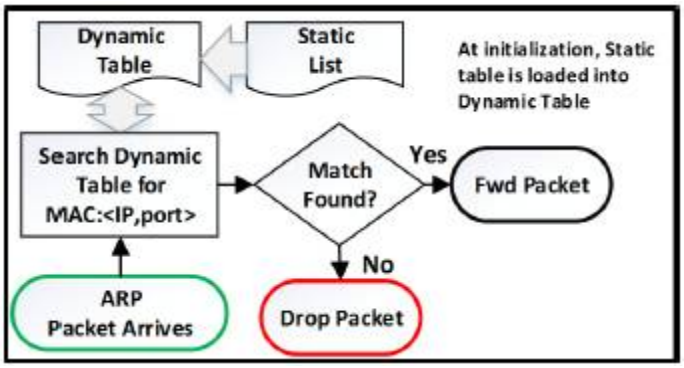

Fig. NFG ARP Validation Flow Graph

While building the dynamic table, NFG utilizes the MAC address of each network device as its primary key in the table entry. The row for each entry contains an IP address, port number, entry state, and static (or fixed port) indication that is specific to its MAC address. For security purposes, entries to the dynamic table can only be initialized by a valid DHCP-offer. Only then can a host's DHCP-request be used to assign a port to an established MAC-IP entry and move the entry's state to pending. Once the DHCP serveresponds with a DHCP-ack, the entry is verified, and that MAC-IP-port association is then allowed to submit ARP-replies. Otherwise, the ARP packet is dropped at the switch. as shown in Fig. 3.7, NFG limits interactions with untrusted devices by only using DHCP-request packets that are bounded by DHCP-offer and DHCP-ack packets DHCP-discovery packets are ignored.

V. MATHEMATICAL MODEL

5.0 NETWORK FLOW GUARD SECURITY

In this section, we discuss the goals, design, and compo-nents for our Network Flow Guard (NFG) security implementation. In doing so, we show how an OpenFlow [54] switch, linked to a Ryu [55] SDN controller, using Ryuretic [10], and working in cooperation with a Trusted Agent

To do this, NFG's TTA analysis module maintains a running average for each switch port by capturing the TTA (time between a destination IP's TCP[SYN,ACK] packet and the client's TCP[ACK] packet) and adds it to the switch port's running average (portAv) Hence, NFG must also maintain a state table for the TCP connections of all client-connected switch ports.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 5, Issue 12, December 2017

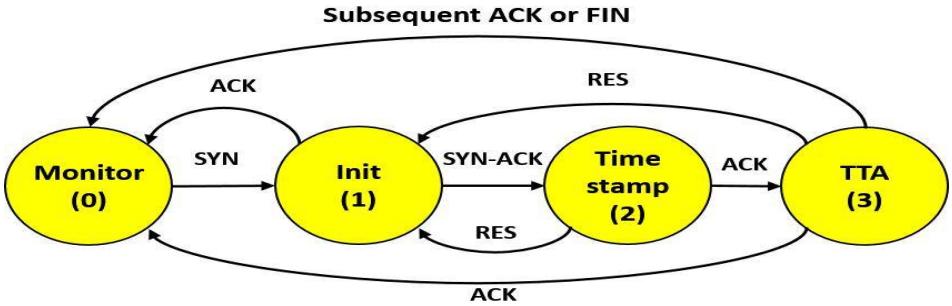

Fig. 2: State Diagram for Capture of Port Flow TTA

The controller monitors for the corresponding client's TCP[ACK] packet and records its timestamp. The difference between timestamps is recorded as a TTA. NFG then removes the table entry for the TCP connection.

- **portAvinit = { [(weight* speed) +TTA]/(Weight +1) }**
- **portAvnext = { [(weight* portAvold + TTA]/(Weight +1) }**

Where,
TTA - Time-to-Acknowledge
PortAv - PortAverage.

Once the TTA is obtained, the value is entered into either Eqn. (1) or Eqn. (2) depending on whether the port average (portAv) is being initialized or not Eqn. (1) calculates an initial port average after the controller obtains its first TTA. This value, seeded with an initial value of 5ms Eqn. (2) calculates subsequent port averages for the switch port. In both equations, either the seed or the old portAv are weighted with a factor of 9 to ensure gradual changes

VI. CONCLUSION

We consider the capabilities of traditional networks to defend against rogue access points (RAPs), and then offer a first-ever, SDN-based approach to detecting and preventing RAPs. An emerging architecture, webRTC, is also exploited as a novel method for detecting subnets hidden behind NAT devices. Hence, our work contributes a new RAP detection method that is applicable to both SDNs and traditional networks. Other contributions of our work includes its use case for the newly developed wireless network emulation framework (i.e., Mininet-WiFi) and its introduction of a new feature enabling Ryuretic (a modular programming language for SDN application development) to simultaneously redirect packets and modify their header fields. Using Mininet-WiFi, we implement a testbed capable of supporting wireless switches, wireless routers, and wireless hosts (stations). In
doing so, our work further validates Mininet-WiFi's viability towards wireless security application development and testing.

REFERENCES

[1] H. Kim, T. Benson, A. Akella, and N. Feamster, "The Evolution of Network Configuration: A Tale of Two Campuses," ACM, Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference, 2011, pp. 499–514.
[2] H. Kim, J. Reich, A. Gupta, M. Shahbaz, N. Feamster, and R. Clark, "Kinetic: Verifiable dynamic network control," Oakland, CA: 12th USENIX Symposium on Networked Systems Design and Implementation (NSDI 15), May 2015, pp. 59–72, ISBN: 978-1-931971-218.
[3] T. Benson, A. Akella, and D. A. Maltz, "Unraveling the complexity of network management.," NSDI, 2009, pp. 335–348.
[4] M. Shahbaz, S. Choi, B. Pfaff, C. Kim, N. Feamster, N. McKeown, and J. Rexford, "Pisces: A programmable, protocol-independent software switch," AT&T Research Academic Summit, Bedminster, NJ, USA, 2016.
[5] J. Cox, S. Donovan, R. Clark, and H. Owen, "Ryuretic: A modular programming language for ryu," IEEE, Military Communications Conference, 2016. MILCOM 2016. IEEE, 2016.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 5, Issue 12, December 2017

[6] N. Feamster, J. Rexford, and E. Zegura, "The road to sdn: An intellectual history of programmable networks," SIGCOMM Comput. Commun. Rev., vol. 44, no. 2, pp. 87–98, Apr. 2014.
[7] A. Aguado, V. Lopez, J. Marhuenda, O´ . G. de Dios, and J. P. Ferna´ndez-Palacios, "Abno: A feasible sdn approach for multi-vendor ip and optical networks," Optical Fiber Communication Conf.(OFC), 2014.
[8] S. Perrin and S. Hubbard, Practical Implementation of SDN & NFV in the WAN, [Online]. Available: http://events.windriver.com/wrcd01/wrcm/2016/08/WP-practical-implementation-of-sdn-nfv-in-thewan.pdf, 2013.
[9] O. N. Fundation, "Software-defined networking: The new norm for networks,"ONF White Paper, 2012.