IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 8, August 2017

Security Analysis on Software Defined Mobile Network

Mr. A. Venugopal M.Sc, M.Phil[1], Anila. N. V[2]

Assistant Professor, Department of Computer Science, Sree Narayana Guru College, Coimbatore, Tamil Nadu, India [1]

M.Phil Scholar, Department of Computer Science, Sree Narayana Guru College, Coimbatore, Tamil Nadu, India [2]

Abstract: Software Defined Mobile Networking (SDMN) has emerged as new network architecture for dealing with network dynamics through software-enabled control. The SDMN is endorsing several new network applications; however the security has become an important concern in every application. This paper provides an extensive analysis on SDMN with several security considerations. This paper provides the introduction of SDMN and the security threats on it. The most common vulnerabilities in SDMN is Denial of Service attacks, Spoofing, Tampering, Repudiation, Information disclosure and access misuse. In this paper, a review on a wide range of SDMN security mechanisms is discussed; this includes Intrusion Detection systems, Intrusion prevention systems, firewalls, access control, deep packet inspection, and policy management. Finally the problem has been identified in the previous works on SDMN security.

Keywords: Software Defined Network, NFV, Security, Mobile Networks, Monitoring, DOS attacks, Spoofing, Wireless Network.

1. INTRODUCTION

At present Internet supported systems like cloud services, social networks and so many changed the network requirements like routing information, bandwidth demand and topology dynamically. Wireless networks have very limited ability to cope up with such frequent changes in terms of resource. To address this issue, Software Defined Mobile Networking (SDMN) [1] has emerged as a new network architecture that allows for more flexibility through software-enabled mobile network control. This is a part of SDN framework. The basic idea is to separate control plane from data plane into a program, called controller, for dynamic orchestration of network components. While SDMN is enabling new network applications, security has become an important concern as security is not yet a built-in feature in the SDMN architecture. Research has shown that various security attacks can be conducted against SDMN through different network components. As SDMN relies on software, code vulnerabilities also have an important impact on SDMN security [2]. Moreover, SDMN offers abundant opportunities for implementing security controls as SDMN controller applications. Such software solutions can enable more flexible security controls in dynamic and virtualized network environments. They provide a practical means for software-defined security control. In this paper, we provide an overview, architecture of the SDN and SDMN with extensive survey on security issues. The security threats in SDMN are analyzed. Many countermeasures were developed to handle the security issues in the SDMN, those techniques are compared with the real-time scenarios. We also review a wide range of SDMN security control applications like as firewalls, Intrusion Detection / Protection System (IDS/IPS), access control, auditing, packet inspection mechanism and policy management. In addition, we discuss several open issues and research topics that worth further investigation.

The rest of the paper is organized as follows. To facilitate discussions on SDMN security, Section 2 briefly introduces the architecture of SDMN. Section 3 reviews SDMN security threats and countermeasures. Section 4 focuses on overall problem definition. Section 5 concludes this paper.

2. SDMN ARCHITECTURE

Software Defined Mobile Networks aims at providing open, centralized, decoupled, programmable, flow-based, and dynamic network communication mechanisms. The SDMN architecture combines the principles of the SDN, Network Function Virtualization (NFV) [3] and cloud computing. This integration provides high flexibility. A consolidated illustration of the SDMN architecture is presented in Figure 1.

Traditional networking components such as switches and routers are vendor specific. They provide limited ability for users to experiment their own networking protocols on live networks with real traffic. With SDMN, developers can develop middle-boxes that interact with the controller and network switches. SDMN shifts networks from IP-based to flow-based management and control.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 8, August 2017

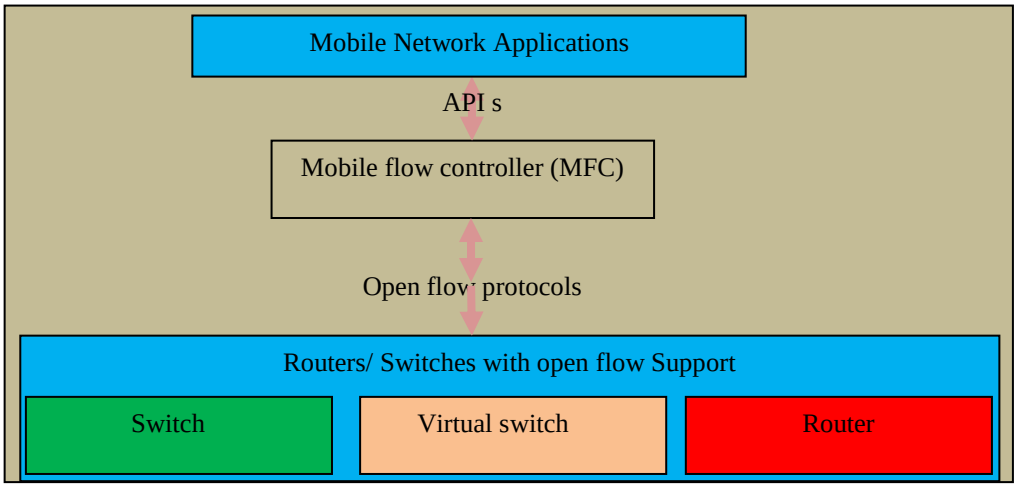

Fig 1.0 architecture of SDMN

SDMN is a flow-based architecture, where forwarding decisions in switches are made according to flows. Records or rules in switches and firewalls are per flow. This will impact many applications that depend on network traffic. For example, typical firewall rules deny or permit packets based on source or destination IP, MAC addresses or ports. Future firewall rules may become more dynamic and be updated frequently based on real time traffic [4]. A major advantage of software over hardware is that it can accommodate frequent changes for more dynamics and flexibility. Configuration or reconfiguration of hardware is labor intensive. Software can be programmed to respond to activities and make decisions dynamically. This is extremely important to those applications with highly dynamic bandwidth demand, such as cloud computing, dynamic datacenters, smart devices, and social networks. Figure 1 shows an overall architecture of SDMN, with the consideration of the most recent technological advances.

3. SECURITY THREATS IN SDMN

Nowadays mobile networks are grown drastically with the help of internet, this nature creates an easy way for several security threats like Denial of Service (DoS) attacks, malware related issues and. The SDN and NFV have their own security limitations. These issues were studied in [5] and [6]. So the deep analysis of security threats on SDMN is mandatory. The mobile network development and deployment on these concepts without analyzing their inherent limitations will create more security challenges. When the integration of NFV and cloud with the SDN, there are more possibilities for the security challenges in the SDMNs. Due to the centralized nature and global visibility the control plane is more vulnerable to security attacks, particularly to DoS and Distributed DoS (DDoS) attacks. Additionally the SDMN affected by many malware and malicious software based issues [7]. The list of security challenges in SDMN is listed in fig 2.0. this section explores the detailed analysis of SDMN security threats. The threats described below can be generic and be applicable to networks in general. As an alternative, threats on SDMN can be classified based on SDMN layers described earlier in Figure 2.0 and the type and nature of attacks that each component can be subjected to. Attacks on SDMN can be also classified based on the type of assets or resources a typical SDMN may have.
For example, attacks can be focused on switches' flow tables where those flow tables include information related to network management; switching, routing and access control. Attacks can be also focused on the controller as the central location for management and control. The channel between the controller and the switches is another major attacks' target where such channel involves important messages that can be hijacked. At the top level, controller communicates with high level applications using a standard interface. Such interface can be also attacked in order to trick the controller to allow malicious applications to join the network and interact with the controller, the network and its traffic.

Application Layer attacks: In the application layer, there are several possible attacks threaten the SDMN. The Authentication and authorization issues will happen in the application layer, which is possible in huge number of third-party apps. The application layer has another issue due to the malicious applications with false flow rules. And the Lack of binding mechanisms for apps also creates the access control related issues.
Control Layer Attacks: Control layer also affected by the several issues like DOS, DDOS, Hijacking, compromising and privacy related attacks. Visible nature of Ctrl-plane generates this kind of attacks. In this type of attacks, there is no compelling mechanism for enforcing access ctrl on backhaul devices. Attacker with access to controller can command to fork any flow at any point to a VNF function anywhere where it can analyze the content breaking confidentiality of communications.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 8, August 2017

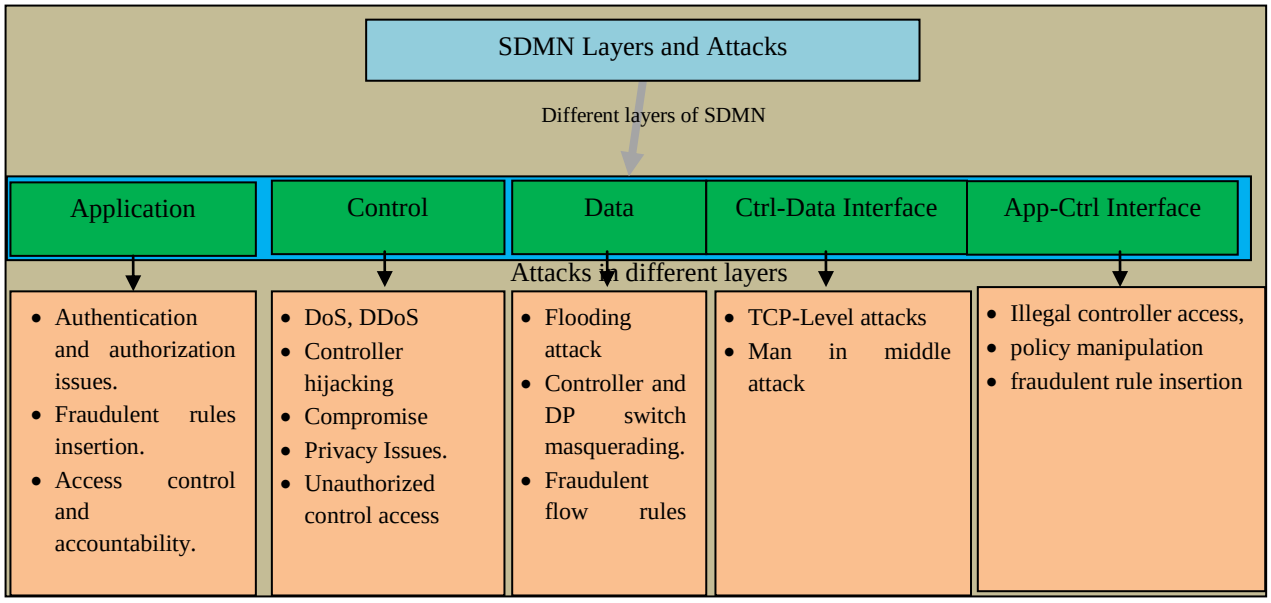

Fig 2.0 Security Challenges In SDMN

Authors in [8] discussed a lightweight method to detect DDoS attacks in SDN. The main challenge was to distinguish normal packets from DDoS flooding packets. They classified network traffic into an attack or normal traffic based on Self Organizing Maps (SOM). The flow features selected were based on earlier approaches studied in [9], including APf (Average Number of Packets in Per Flow (ANPPF), Average of Bytes per flow (ABf), Average of Duration per flow (ADf), Percentage of Pair-flows (PPf) and Growth of Single-flows (GSf). Those metrics or attributes are continuously collected and monitored for detection of possible DDoS. A major concern is that monitoring and maintaining such huge amount of data will significantly degrade controller performance which is already overwhelmed with other tasks. Having a dedicated separate module or controller to perform such task can be a more realistic solution. In paper [10], authors proposed to sample traffic to reduce controller traffic overhead from the monitoring process specified on what can be considered large or abnormal traffic. Once this threshold is exceeded, DoS occurrence can be triggered and controller inserts a flow rule to drop packets. Similarly, traffic map or patterns can be analysed frequently to predict if some traffic is abnormal or large. Authors in [11] proposed content based networking architecture. Controller triggers DoS alert if traffic exceeds a certain threshold. Rules are then inserted in switches by the controller to eliminate source of DoS. In paper [12] an OpenFlow based approach to detect and mitigate botnets is proposed. Botnets are networks or groups of compromised hosts that are used to launch attacks such as DDoS, to propagate worms or send spams. Their proposed solution, COFFEE, utilizes SDN ability to have access to all traffic to reduce rate of false detections. In TCP connections, acknowledgement message (TCP ACK) is required to verify communication between senders and receivers. However, it can also be triggered by a flooding or DoS attack. In Paper [13] authors proposed a simple algorithm to handle TCP ACK packets. A security layer or interface to coordinate the communication between OpenFlow switches and the controller is proposed. A show case of TCP SYN DoS attack is used to evaluate the model. Attack includes occupying all packets and IP address possible combinations. Network performance is measured through the attack to evaluate the time it takes the network to figure and clear out the attack.
In paper [14] authors evaluated an OpenFlow vulnerabilities for DoS and integrity attacks. They showed that OpenFlow protocol and its communication mechanism between controller and switches should be thoroughly investigated. They also conducted an experiment to simulate DoS attacks on Floodlight controller using methods such as TCP SYN or ARP cache poisoning. A vulnerability discovered in Floodlight that disconnects an old switch if a new switch is registered with the same data path ID (DPID) as of the old one. Such vulnerability can be used by malicious switches to claim to be legitimate. The only information attackers need is the DPID which can be acquired from the controller REST API. Auhthors in [15] presented a simple use case for using sFlow monitoring tool for DDoS attacks' detection in OpenFlow. The goal was to counter DDoS without disrupting normal traffic. They used the module "static flow pusher" from Floodlight controller and claimed that no commercial virtual switch showed the same expected response as the open source virtual switch (vSwitch). In paper [16] authors proposed an autonomic DDoS detection system based on OpenFlow. The system uses the simple volume count (i.e. flows/packets per time) to judge the occurrence of DoS or DDoS. The problem with such simple metric is that many false positive alarms may occur where large volume traffic can be legitimate. While some studies argued that OpenFlow networks have more problems with DoS than traditional networks, They showed that SDN can produce a better way of handling Remote Triggered Black Hole (RTBH). This is a technique in traditional WAN networks to countermeasure DoS attacks by instructing routers to

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 8, August 2017

drop all traffic to the target. They used OpenFlow traffic flow statistics to monitor traffic volume and alert for a significant increase in size attributes (e.g. byte and packet counters). They used the mathematical standard deviation measure to evaluate whether certain flows are significantly above average. Packet symmetry is also used as an indicator of DoS in that the difference between incoming and outgoing flows for a particular host is very high. DoS can be handled by effective and dynamic response methods to handle occurrences of DoS. Rate or limit traffic by the controller and monitor abnormal traffic behaviours are also important responses. There are some proposals for active countermeasures of DoS or flooding attacks in SDN networks specifically. Active response means to take an offensive action to counter an attack. An attacker can focus DoS on the messages from data plane or switches to the controller and try to saturate both switch flow table and controller resources; data-to-control saturation attacks. Protection mechanisms should ensure that controller and switches have the ability to quickly recover from such flooding.
IP spoofing is usually used as an opening to other types of security attacks, such as DNS tampering or amplification. A DNS is a directory that associates IP addresses to domain names. To reroute traffic to illegitimate websites, an attacker may manipulate DNS directory. This can be also part of a large flooding or worm spreading attacks. What all spoofing methods have in common is that they try to redirect traffic to illegitimate hosts. They can also be considered to achieve Man in the Middle (MiM) attacks. Spoofing can be mitigated by a proper authentication scheme. Strong password and encryption methods should be enforced to avoid unauthenticated intrusion.

Table 1.0 The Security Mechanisms For Different Layers

Layers Paper Id	**Application**	**Control**	**Data**	**Ctrl-Data Interface**	**App-Ctrl Interface**
17	Threat detection and mitigation				Threat detection and mitigation
18	App debugging, flow rules inspection		App debugging, flow rules inspection		
19,20		Flow rules verification, Configuration verification	Flow rules verification, Configuration verification		
21	Flow policy verification, catch bugs in of programs	Flow policy verification, catch bugs in OF programs			
22	App testing and debugging				
23				Conflict resolution, authorization, security audit system	Conflict resolution, authorization, security audit system
24,25				DDoS detection, Controller resilience	
26			Link monitoring	Link monitoring	
27		Find contradictions in flow rules, authorize applications	Find contradictions in flow rules, authorize applications		
28,29	Controller availability, network monitoring			Controller availability, network monitoring	
30		Access control and dynamic policy enforcement	Access control and dynamic policy enforcement		

The table 1.0 shows the security architectures for the different layers of SDMN. There are several types of solutions are common for different layers.

IV. CONCLUSION

The paper presented an overview of the existing research in SDMN security, focusing on security threats, and security

controls. It is important to note that the landscape of SDMN security changes with the advances in SDMN research and development. For instance, a new protocol or API introduced to SDMN may incur particular security threats and thus require specific countermeasures. To conclude this paper, we discuss several SDMN security issues and research topics. While many existing work suggest directions of further research, they are as follows; detecting different types of new attacks like insider attack, virtual attacks by enforcing policy life cycle, on demand security services, security embedding routing etc.,

REFERENCES

[1]. Liyanage, Madhusanka, Andrei Gurtov, and Mika Ylianttila, eds. Software Defined Mobile Networks (SDMN): Beyond LTE Network Architecture. John Wiley & Sons, 2015.
[2]. Chen, Min, Yongfeng Qian, Shiwen Mao, Wan Tang, and Ximin Yang. "Software-defined mobile networks security." Mobile Networks and Applications 21, no. 5 (2016): 729-743.
[3]. H. Hawilo, A. Shami, M. Mirahmadi, and R. Asal, "NFV: State of the Art, Challenges and Implementation in Next Generation Mobile Networks (vEPC)," arXiv preprint arXiv:1409.4149, 2014.
[4]. Wang, Juan, Wang Yong, Hu Hongxin, Sun Qingxin, Shi He and Zeng Longjie, Towards a Security-Enhanced Firewall Application for OpenFlow Networks. CSS 2013, p. 92-103.
[5]. M. Liyanage, A. Abro, M. Ylianttila, and A. Gurtov, "Opportunities and Challenges of Software-Defined Mobile Networks in Network Security Perspective," IEEE Security and Privacy Magazine, 2016.
[6]. I. Ahmad, S. Namal, M. Ylianttila, and A. Gurtov, "Security in Software Defined Networks: A Survey," IEEE Communications Surveys & Tutorials, vol. 17, no. 4, pp. 2317–2346, 2015.
[7]. A. Y. Ding, J. Crowcroft, S. Tarkoma, and H. Flinck, "Software Defined Networking for Security Enhancement in Wireless Mobile Networks," Computer Networks, vol. 66, pp. 94–101, 2014.
[8]. Braga, Rodrigo and Mota Edjard and Passito Alexandre, 2010, Lightweight DDoS Flooding Attack Detection Using NOX/OpenFlow, In Proceedings of the IEEE Conference on Local Computer Networks (LCN), Denver, CO, USA, 11–14 October 2010, p. 408–415.
[9]. Feng, Yifu, Wang Dongqi and Zhang Bencheng, Research on the Active DDoS Filtering Algorithm Based on IP Flow, In 2009 Fifth International Conference on Natural Computation. IEEE, 2009, p. 628–632.
[10]. Shirali-Shahreza and Ganjali Sajad, Yashar, FleXam: Flexible Sampling Extension for Monitoring and Security Applications in OpenFlow, HotSDN'13, August 16, 2013b, Hong Kong, China.
[11]. Suh, Michelle, Park Sae Hyong, Lee Byungjoon and Yang Sunhee, Building Firewall over the Software-Defined Network Controller, February 16~19, ICACT2014, 2014.
[12]. Schehlmann, Lisa, and Baier Harald, COFFEE: a Concept based on OpenFlow to Filter and Erase Events of botnet activity at high-speed nodes. In INFORMATIK 2013
[13]. Shin, Seungwon and Gu Guofei, Attacking Software Defined Networks: A First Feasibility Study, In Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking, 2013, p. 165-166.
[14]. Benton, Kevin Camp Jean, and Small Chris, OpenFlow Vulnerability Assessment, In Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking. ACM, 2013, p. 151–152.
[15]. Yuzawa, Tamihiro, OpenFlow 1.0 Actual Use-Case: RTBH of DDoS Traffic While Keeping the Target Online, <http://packetpushers.net/openflow-1-0-actual-use-case-rtbh-of-ddos-traffic-while-keeping-the-target-online>, 2013.
[16]. YuHunag, Chu, Tseng Min-Chi, Cen YaoTing, Chou YuChieh, and Chen YanRen, A Novel Design for Future On-Demand Service and Security. In Proceedings of the International Conference on Communication Technology (ICCT), Nanjing, China, 11–14 November 2010, p. 385–388
[17]. S. Shin, P. A. Porras, V. Yegneswaran, M. W. Fong, G. Gu, and M. Tyson, "FRESCO: Modular Composable Security Services for Software-Defined Networks." in NDSS, 2013.
[18]. R. Beckett, X. K. Zou, S. Zhang, S. Malik, J. Rexford, and D. Walker, "An Assertion Language for Debugging SDN Applications," in Proceedings of the Third Workshop on Hot Topics in Software Defined Networking, ser. HotSDN, vol. 14, 2014, pp. 91–96.
[19]. [19] A. Khurshid, W. Zhou, M. Caesar, and P. Godfrey, "Veriflow: Verifying Network-wide Invariants in Real Time," ACM SIGCOMM Computer Communication Review, vol. 42, no. 4, pp. 467–472, 2012.
[20]. [20] E. Al-Shaer and S. Al-Haj, "FlowChecker: Configuration Analysis and Verification of Federated OpenFlow Infrastructures," in Proceedings of the 3rd ACM workshop on Assurable and usable security configuration.
[21]. 21] S. Son, S. Shin, V. Yegneswaran, P. Porras, and G. Gu, "Model Checking Invariant Security Properties in OpenFlow," in Communications (ICC), 2013 IEEE International Conference on. IEEE, 2013, pp. 1974–1979.
[22]. M. Canini, D. Kostic, J. Rexford, and D. Venzano, "Automating the Testing of OpenFlow Applications," in Proceedings of the 1st International Workshop on Rigorous Protocol Engineering (WRiPE), no. EPFLCONF- 167777, 2011.
[23]. Security-Enhanced Floodlight. [Online]. Available: http://www. sdncentral.com/education/toward-secure-sdn- controllayer/,2013/10/
[24]. R. Braga, E. Mota, and A. Passito, "Lightweight DDoS Flooding Attack Detection using NOX/OpenFlow," in Local Computer Networks (LCN), 2010 IEEE 35th Conference on. IEEE, 2010, pp. 408–415.
[25]. P. Fonseca, R. Bennesby, E. Mota, and A. Passito, "A Replication Component for Resilient OpenFlow-based Networking," in Network Operations and Management Symposium (NOMS), 2012 IEEE. IEEE, 2012, pp. 933–939.
[26]. J. Kempf, E. Bellagamba, A. Kern, D. Jocha, A. Tak´acs, and P. Skoldstrom, "Scalable Fault Management for OpenFlow," in Communications (ICC), 2012 IEEE International Conference on. IEEE, 2012, pp. 6606–6610.
[27]. P. Porras, S. Shin, V. Yegneswaran, M. Fong, M. Tyson, and G. Gu, "A Security Enforcement Kernel for OpenFlow Networks," in Proceedings of the first workshop on Hot topics in software defined networks. ACM, 2012, pp. 121–126.
[28]. K. Phemius, M. Bouet, and J. Leguay, "Disco: Distributed Multi-domain SDN Controllers," in Network Operations and Management Symposium (NOMS), 2014 IEEE. IEEE, 2014, pp. 1–4.
[29]. Y. Zhang, N. Beheshti, and M. Tatipamula, "On Resilience of Split- Architecture Networks," in Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE. IEEE, 2011, pp. 1–6.
[30]. A. K. Nayak, A. Reimers, N. Feamster, and R. Clark, "Resonance: Dynamic Access Control for Enterprise Networks," in Proceedings of the 1st ACM workshop on Research on enterprise networking. ACM, 2009, pp. 11–18.
[31]. H. Hu, W. Han, G.-J. Ahn, and Z. Zhao, "FLOWGUARD: Building Robust Firewalls for Software-Defined Networks," in Proceedings of the third workshop on Hot topics in software defined networking. ACM, 2014, pp. 97–102. ACM, 2010, pp. 37–44.