# A Novel Design for Real-Time Intrusion Response in Latest Software-Defined Networks by Graphical Security Models


**L. Sri Ramachandra and K. Hareesh**

**Abstract** In the current era, many of the application domains are adopting software-defined network (SDN). SDN provides a special functionality to the network flow by controlling it dynamically with more robustness. Traditional networks are high in cost whereas SDN is economical. Since there are more chances of cyber-attacks, security solutions are proposed to reinforce and retreat the SDN. These security solutions have to be compared with each other and select the optimized solution to provide the best security for SDN. Graphical security models like attack trees and attack graphs can be used to measure and estimate the safety of SDN. Due to computational complexity, it is hard to provide security for SDN against cyber-attacks in real-time cases. Using precomputations, this paper aims to detect the disturbance which causes in SDN and finds the possibility of future attacks in the future for real-time cases. Various SDN components are taken into considerations for conducting an assessment on security which was not accessible in the existing network. Experimental analysis of this paper estimates all the possible attacking path in an ongoing attack which can be mitigated in real-time cases. It also exposes the security metrics which depends on the flow table that includes SDN components. Hence, it is possible to provide security for SDN against cyber-attacks in real-time cases.

**Keywords** Software-defined networks (SDN) · Moving target defense (MTD) systems · Attack graph · Hierarchical attack representation model (HARM) · Remote triggered black hole (RTBH)

L. S. Ramachandra (✉)
Department of Computer Science and Engineering, Government Engineering College, Ramanagara, Karnataka, India
e-mail: dsram49@gmail.com

Visvesvaraya Technological University, Belagavi, Karnataka, India

K. Hareesh
Department of Computer Science and Engineering, Government Engineering College, K R Pet, Karnataka, India
e-mail: hareeshk.gec@gmail.com



P. Karuppusamy et al. (eds.), *Sustainable Communication Networks and Application*, Lecture Notes on Data Engineering and Communications Technologies 55,
https://doi.org/10.1007/978-981-15-8677-4_45

## 1 Introduction

Logical network topology can be dynamically changed in real time by administrators of network if and only if SDN permits [1]. This can be achieved when controls are separated from data flows on top of the data plane and control plane. When network topology [2] is reconfigured dynamically by SDN, impact on performance by network disruptions will be negligible. This efficiently helps in optimization of load by administrators in real-time cases. New architecture like moving target defense systems (MTD) has to be designed and deployed to provide security for SDN. New networking components like forwarding devices and controllers are introduced by SDN which provides attackers with an attack vector for exploiting the SDN. Some of the security techniques such as DELTA and Athena are developed to protect SDN from cyber-attacks. DELTA [3] is a framework for evaluating the security, whereas Athena [4] is a framework developed for anomaly detection. Athena uses machine learning for predicting the patterns of various attacks. Graphical security models will be used for the evaluation of the SDN. Attack trees (AT) and attack graphs (AG) are used as security models since they provide security in-depth and compute the optimal countermeasures. New SDN components have to be considered for the assessment of security when this approach is applied to SDN. As there will be a delay in initial attack and response, this causes difficulty for intrusion detection system to find out the ongoing attacks in real time. This eases the attacker to reach the target so there should be an effective focus on a countermeasure to avoid the attacker reaching the target than giving more importance in finding out the attack detection. All the probable attacking paths have to be predicted to know the targets of the attacker. Computing all these possible attacking paths will lead to adaptability and scalability problems [5, 6]. To avoid all these problems, an efficient technique is required which can take care of all the attacking paths meanwhile considering new SDN components for the evaluation of the security. To overcome the above-mentioned problems, the hierarchical attack representation model (HARM), a graphical security model is incorporated with the precomputed approach along with the components of SDN. This evaluates the security of SDN in real-time cases. HARM helps in evaluating all the possible attacking paths before an attack is detected. This helps in formulating effective countermeasures and helps in estimating the possible attacking paths from the detection point. Evaluation is carried out by generating precomputed possible scenarios of attack with the help of a full attack graph (AG).To know how an attacker is trying to steal data from the outside, a scenario of attack was used. After identifying the paths of attack, it is found why the intrusion detection system took a long time to detect the attack than the time of the attack. To identify the relevant paths of attack, full attack graph which is precomputed is used. By evaluating this full AG, related countermeasures are deployed. Some of the experimental analyses are conducted to demonstrate that the proposed approach can efficiently detect the attacks without any delay and can diminish the ongoing current attack in real-time cases.

The contribution of the paper are summarized as follows:

- When new SDN components are taken into account with their respective attack vectors, security assessment is conducted for SDN.
- To develop a countermeasure for SDN and to provide a security assessment for real-time cases, precomputed attack scenarios are generated using full attack graph.
- To avoid the delays in finding an attack in an ongoing intrusion, response and prevention mechanisms are proposed in attack detection systems.
- Experimental analysis is conducted for the demonstration of the proposed approach for resolving the attacks in SDN with delayed detection.

The organization of the paper is as follows, related work is presented in Sect. 2, and a framework is presented in Sect. 3. Real-time intrusion response in SDN is presented in Sect. 4. Precomputation and attack prediction for security assessment are presented in Sect. 5. Section 6 presents results and analysis and finally Sect. 7 presents conclusion of this work.

## 2 Related Work

- SDN security: There are certain security issues to SDN which are discussed in [1, 7–10]. Threat vectors of SDN were presented by Kreutz et al. which were not there in the traditional network system. Potential solutions were given to resolve the threats which were identified but a method for evaluating those threads was not mentioned. The paper aims to provide a solution to these problems. For the better enhancement of OpenFlow protocol security in SDN, a framework called FRESCO is presented by Shin et al. [11] which allows SDN to detect and resolve the attacks. This work demonstrates that SDN switches which are the part of SDN components are more disposed to cyber-attacks. SE-Floodlight was presented by Porras et al. [12] which was the extension of OpenFlow Floodlight controller. This protects the SDN control plane with added security features.
- SDN intrusion detection: An attack which is carried out successfully is defined to be intrusion which includes malicious activities in the system. It will be perfect if all the intrusions are detected with the accuracy of 100% but in practice, and it is not feasible. This is because of the true positive and false positive alarms of the intrusion detection systems. A framework was proposed by Dhawan et al. [13] which helps to detect attacks in data plane forwarding and network topology. Based on capabilities on traffic flow, detecting distributed denial of service (DDoS), a lightweight method is proposed by Braga et al. [14]. OpenFlow protocol was used to advance the features of remote triggered black hole (RTBH). This can mitigate the distributed denial of service upon apply of SDN. This is demonstrated by Giotis et al. [15]. By using an efficient mechanism, anomaly detection and mitigation were performed in the architecture of SDN which is presented in the paper.

- SDN security modeling: A selection framework NICE was presented by Chung et al. [16]. This security model was developed as an intrusion detection system in the network and as a countermeasure framework. Attack graph was used for the evaluation of security which was a core work of the framework. Usage of attack graph is limited because of its scalable difficulties. In the same way, different graphical security models grieve from adaptability and scalability problems. The constraints in real time are the typical reason for the problems like scalability. To overcome all these issues, our approach suggests that attack scenarios have to be precomputed in advance and use them whenever it is necessary. Therefore, a full attack graph is used to create the possible precomputed attack paths. To address the adaptability and scalable problems, HARM is also used in our approach. When there is a detection of intrusion in a network system and effective countermeasure is formulated in real time, it is possible to evaluate security position of the SDN quickly with the precomputation of all the possible attacking paths.

## 3 A Framework for Real-Time Intrusion Response in SDN

A graphical security model is proposed to solve the problems of IDSes. The security model is precomputed. This model is used in the real-time interruption reaction in SDN. Some of the general steps are as follows:

- Information regarding vulnerabilities in security and dependencies in node connection of SDN has to be collected which are related with configuration of SDN.
- GSM for security valuations are generated by gathering all the collected inputs.
- Data on intrusion detection from SDN will be collected.
- Effective attack response is calculated with the selection of optimum countermeasure.

Figure 1 represents the relationship between the above steps mentioned.

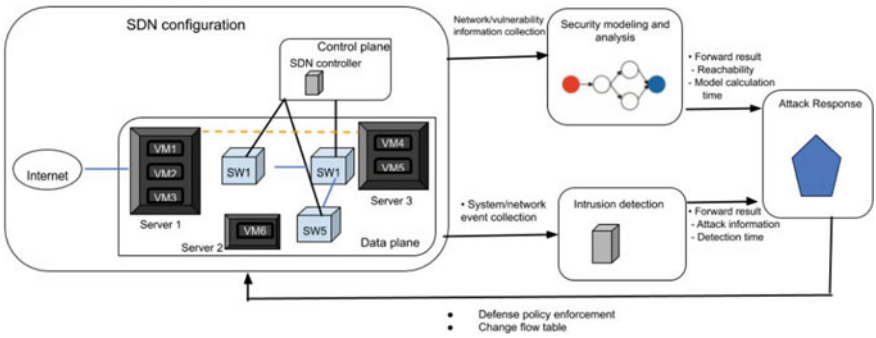


**Fig. 1** Framework for real-time intrusion response in SDN

### 3.1 SDN Configuration

Necessary information regarding security which is associated with SDN has to be collected. Connectivity of each SDN component along with its respective dependencies and vulnerability of each component of SDN are the two main information. NESSUS and OpenVAS are the two vulnerability scanning tools used to collect the information of vulnerability associated with SDN components. With the help of settings in the SDN controller and flow table, dependencies of the components can be collected. Later the module, security modeling and analysis retrieve this information. IDSes are also present in SDN. If any of the intrusions are detected, those are directly sent to the intrusion detection module. Predicting the attack scenarios become more complex in nature as IDSes are not placed in every SDN components.

Figure 1 represents the framework for real-time intrusion response in SDN. In the SDN configuration module, SW acts as a connection between the data plane and the control plane. Six virtual machines are used which have connectivity to the Internet. All these virtual machines are placed in the server. SDN controller is placed inside the control plane. Security modeling and analysis module retrieve the information from the SDN configuration system to the network-related issues and vulnerabilities.

### 3.2 Security Modeling and Analysis

GSM can be generated with the inputs taken from the module of SDN configuration. For example, a model which is scalable and adaptable is HARM model which is used for demonstration purpose in this paper. Any of the GSM models can be used for the demonstration purpose but a HARM model has been chosen. Different security vulnerabilities are considered by GSM, and it computes diverse attack scenarios which have various dependencies. When the network size increases, the scalability problem also increases. Therefore, to achieve the response in the real-time attack, a technique of precomputation is required. Then, the module of attack response receives the information of precomputed assessment of security. This information will be used when there is a detection of intrusion.

### 3.3 Intrusion Detection

Intrusion logs are collected from the IDSes of SDN which will be then sent to the attack response module. Raw data regarding intrusion detection is processed in this module. Attack information such as attack type and metadata such as time of attack and location will be analyzed. This module will try its best to detect the attack fast and accurately. But in real time, its performance cannot be completely depend on.

### 3.4 Attack Response

This module is the heart of the architecture. Attack impact will be evaluated in this module by considering the time of intrusion detection and attack location. More than one computer will be affected if an attack takes place in a situation where a subnet is located. This module aims at reducing the attack impact by damage estimation, detecting the location of an attack and isolates the attack from getting progressed.

## 4 Real-Time Intrusion Response in SDN

### 4.1 SDN Configuration

By considering an example shown in Fig. 2, the usability of the proposed solution is demonstrated.

Nine nodes are included in the example of the toy where three switches and six virtual machines are considered. Assumed that the location of an attacker to be outside the SDN and Internet connection is provided only to the virtual machines that are located on the Web server. Since our proposed system takes care of both IDSes and security, the attackers who are inside the SDN can also be focused. Virtual machines aim at providing services to those who are located within as well as externally located. If a system does not encounter any problem, it works as follows. By considering the scenario from Fig. 2, when a user requests a data that is stored in the database (Virtual Machine 6), the virtual machine which is stored in the Web server (Virtual Machine 1) sends a request for the virtual machine in the application server (Virtual Machine

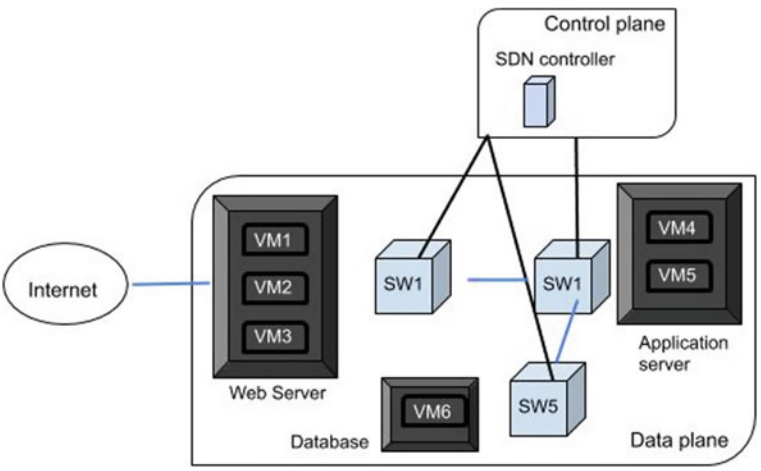


**Fig. 2** SDN configuration example

4). From the application server, the requested data will be forwarded to the database for further processing. After processing the data, it is sent to the user. This procedure is followed only for the valid requests from the user. In this example, the data is processed from Virtual Machine 1 → Virtual Machine 4 → Virtual Machine 4.

Sometimes it leads to burst time when there are more requests from the user. Redundancies will be created to provide service in such case of emergency. In Fig. 2, redundancy is created between VM1 and SW2, another redundancy is between VM5 and SW1. If the attacker is successful in breaking the SDN, various attack paths will be generated using these redundancy connections.

## *4.2 Security Modeling and Analysis*

### 4.2.1 Common Vulnerability Scoring System (CVSS)

The base score of CVSS is used for measuring the vulnerability severities. Physical metric, user interaction, and privileges required are added to the attack vector by considering the base vectors. The rank of integrity is changed from partial to low, availability rank is changed from complete to high whereas the rank of confidentiality remains same to be none. Probabilities of success of an attack and its impact have to be known for computing the security risk. The success of attack probability is represented using the exploitability metric which is associated with each vulnerability.

### 4.2.2 Attack Graph for SDN

Arbitrary code is executed on Virtual Machine 6. The definition of attack graph is as follows, an attack graph is a directed graph such that AG = $(V, E)$, where $V$ is the finite set of vulnerabilities present in the network system and $E$ is the set of edges. Figure 3 represents the attack graph which is generated for mapping the various attack scenarios.

## *4.3 Intrusion Detection*

Here, time is taken into consideration when an attack is detected. There will be chances of getting progress in the attack while the attack is getting detected. Therefore, considering the proper attack scenario plays a major role to mitigate the attack. The Bayesian theory has to be considered for the detection of an attack but here it is assumed that SDN mechanisms are correct for detection of an attack. Threshold random walk which is associated with the credit-based algorithm is similar to that of

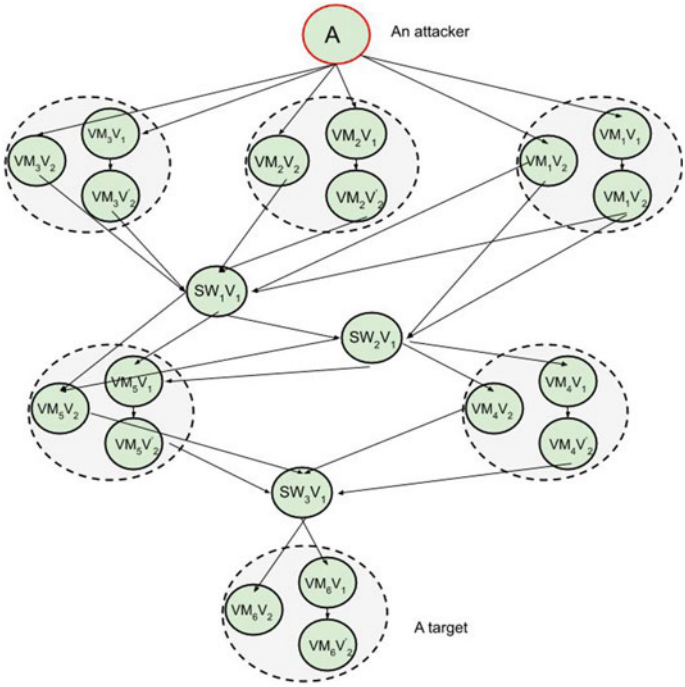


**Fig. 3** An attack graph of SDN

applying Bayesian theory. If detection of an attack is delayed, then the attacker will be much progressed than the prediction. This scenario is represented in Fig. 4.

Figure 4 represents the location of the attacker at SW1 with the red dotted circle. Here, the attacker has successfully reached SW1 after passing through VM2. SDN administrator has been alerted only about the attack at VM2 and not SW1. To overcome this problem, the full attack graph is used which can be used to predict all the possible attack paths. This is represented in Fig. 4. As a countermeasure for the problem, flow table rule change is used. This limits the attack path from the number of hops from the node where the initial detection has taken place. For example, if the number of hops considered is 2, then the results are represented in Fig. 4. By following this, the attacker can be blocked by further accessing the paths. This proves that SDN functionalities can be maintained along with disconnecting the ongoing attacks.

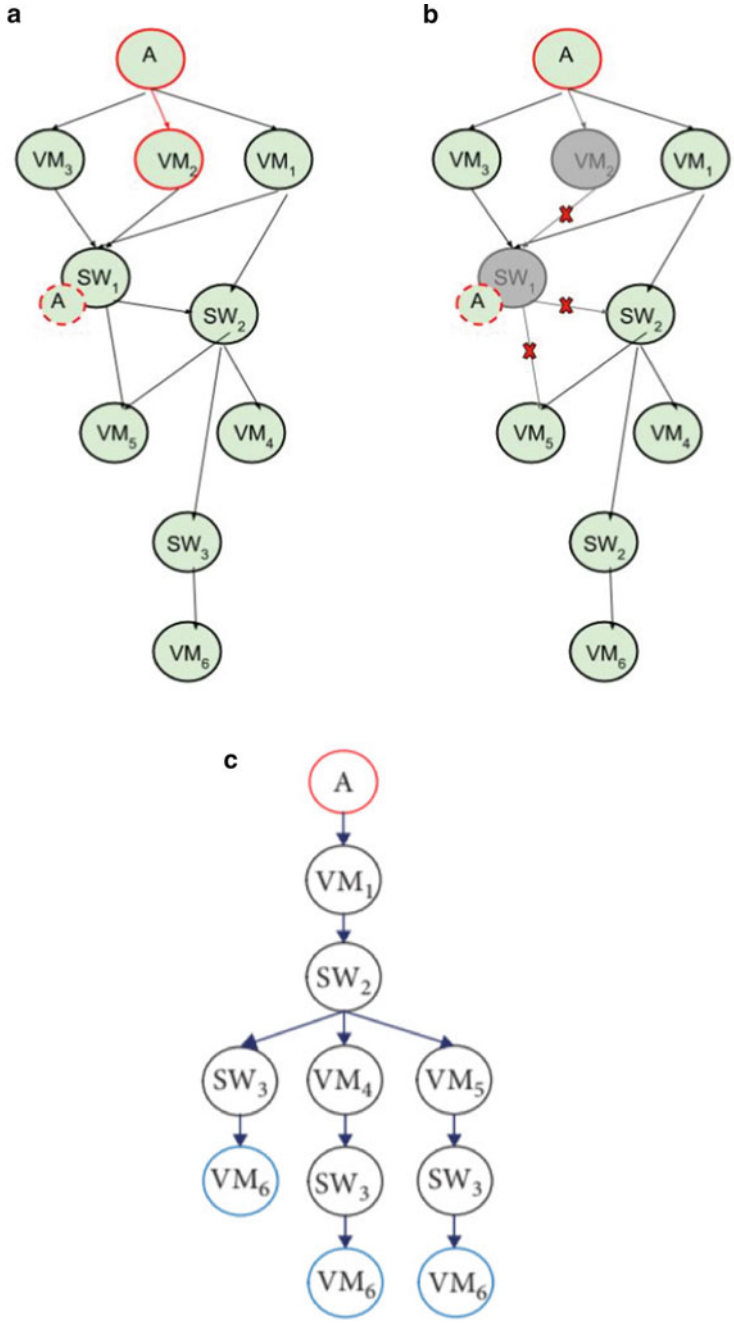


**Fig. 4** **a** Representation of attack detection figures, **b** applying flow at Virtual Machine 2 table to block VM2, and **c** apply of full attack graph for countermeasure

## 5 Precomputation and Attack Prediction for Security Assessment

By considering the delays that occur in the detection of an attack, precomputation and prediction of attack are introduced. By precomputing the attack scenarios, the time taken to detect the attack can be reduced. Defense mechanism for SDN is enhanced to avoid the delay in finding the attack. Full graph and HARM are explained in this session.

- Full graph: Graphical security models cause a scalability problem while conducting the assessment of SDN security. Full attack graph is used to precompute the scenarios of all possible attack paths. This precomputed offline information can be later used in real-time cases whenever it is necessary.
- HARM: The above explained full attack graph is used in security assessment for particular attack scenarios of fast real-time attacks. In this case, the security problem arises since computing all the attack scenarios is not scalable. To provide more scalability, HARM is used through which SDN security is assessed. To reduce the scalability problem, network nodes of HARM are split into multiple layers.

## 6 Result and Analysis

By considering different security metrics, full attack graph is evaluated to check its effectiveness for precomputation. Any of the models can be used but the security metrics will not be changed. Since the computation results, full attack graph and HARM are same, and its result is not explicitly mentioned in this paper. Changes are observed in the security metrics by deploying the countermeasure and also without deploying it. Flow table rules are also changed to block the attack paths. Simulations are conducted to observe the difference between the performance of a full attack graph and HARM.

- Change in security metrics: Fig. 2 is used as the experimental scenario for the investigation in this session. Services are not available in the system until and unless the database receives the request. Therefore, the flow table rules of SW3 and Virtual Machine 6 are not changed by network administrator because of the system constraints. At least one path exists to connect the user requests to the network by ensuring the operability is made sure that. Minimal cost is considered for the improvement of security of SDN, by doing so, the performance of SDN will not be affected when the flows are modified.
- Numerical sensitivity analysis: Attacker can attack the node very fast when the response to the attack is very slow which results in the system loss. Comparison is made for the loss occurred due to slow response to an attack and to the cost required to respond to an attack in a fast way. The sensitive analysis method is applied since cost loss and cost spent to respond to an attack cannot be defined.

Response cost and attack cost are calculated based on attack time and the time taken for attack detection.

- Simulation: Evaluation time and generation time are simulated to examine the performance comparison of full attack graph precomputation to that of simple attack graph. Full attack graph precomputation is very important as it reduces the detection time in real ongoing attack and also reduces the evaluation time for security.

## 7 Conclusion

Network communications can be managed economically and more robustly by the SDN functionalities controls the network flow dynamically. New attack vectors and vulnerabilities were introduced which were not present earlier. Many security solutions were proposed for SDN to strengthen its power against the cyber-attacks. But still, the security problem of SDN and its functionalities are not solved which makes system administrator ensure a regular systematic check for the system security.

A framework for system modeling and Analysis was proposed in this paper which ensures the countermeasure for a real ongoing attack with the generation of precomputation for SDN. For the generation of possible attack scenarios to the current SDN, attack graph, full attack graph, and HARM use the precomputation method. These results are used as a combination with the precomputed attack scenarios of IDS and correlate the computation. When an attack detection mechanism fails, these models are later used to detect the potential attacks in real-time cases. To verify this, experiments are conducted and simulations were made on the SDN testbed which proved that our approach can be effectively used in the attack detection in a real-time ongoing attack and defend the attack.

**Acknowledgements** This research was supported by Visvesvaraya Technological University , Belagavi, Karnataka, India. We thank our principal and colleagues from Government Engineering College, Ramanagara who provided insight and expertise that greatly assisted the research and for assistance with particular technique for comments that greatly improved the manuscript.

## References

1. Stabler, G., Rosen, A., Goasguen, S., Wang, K.-C.: Elastic ip and security groups implementation using openflow. In Proceedings of the 6th International Workshop on VirtualizationTechnologies in Distributed Computing Date, ser. VTDC'12, pp. 53–60, ACM, New York, NY, USA (2012)
2. Tariq, M., Koldehofe, B., Bhowmik, S., Rothermel, K.: PLEROMA: a SDN-based high performance publish/subscribe middleware. In Proceedings of the 15th InternationalMiddleware Conference (Middleware 2014), pp. 217–228, ACM, New York, NY, USA (2014)

3. Lee, S., Yoon, C., Lee, C., Shin, S., Yegneswaran, V., Porras, P.A.: Delta: a security assessment framework forsoftware-defined networks. In Proceedings of the 2017 Network and Distributed System Security Symposium, San Diego,CA, USA, March 2017
4. Lee, S., Kim, J., Shin, S., Porras, P., Yegneswaran, V.: Athena: a framework for scalable anomaly detection in software-defined networks. In Proceedings of the 2017 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), pp. 249–260, Denver, CO, USA, June 2017
5. Hong, J., Kim, D.: Performance analysis of scalable attack representation models. In: Janczewski, L., Wolfe, H., Shenoi, S. (eds.) Security and Privacy Protection inInformation Processing Systems (SEC 2013), vol. 405, pp. 330–343. Springer, Berlin, Germany (2013)
6. Lippmann, R., Ingols, K.: An Annotated Review of Past Papers on Attack Graphs, Technical report ESC-TR-2005- 054. MIT Lincoln Laboratory, Lexington, MA, USA (2005)
7. Gude, N., Koponen, T., Pettit, J., et al.: Nox: towards an operating system for networks. ACM SIGCOMM Comput. Commun. Rev. **38**(3), 105–110 (2008)
8. Casado, M., Garfinkel, T., Akella, A., et al.: A protection architecture for enterprise networks. In: Proceedings of the 15$^{th}$Conference on USENIX Security Symposium-Volume 15, ser.USENIX-SS'06, USENIX Association, Berkeley, CA, USA, July 2006
9. Matias, J., Garay, J., Mendiola, A., Toledo, N., Jacob, E.: Flownac: flow-based network access control. In Proceedingsof the 2014 'ird European Workshop on Software DefinedNetworks, ser. EWSDN'14, pp. 79–84, IEEE Computer Society, Washington, DC, USA, September 2014
10. Guang Yao, J.B., Xiao, P.: Source address validation solution with openflow/nox architecture. In: Proceedings ofthe 2011 19th IEEE International Conference on NetworkProtocols solution with openflow/nox architecture, pp. 7–12, Vancouver, Canada, October 2011
11. Shin, S., Porras, P.A., Yegneswaran, V., et al.: Modular composable security services for software-defined networks. In: Proceedings of the 20th Annual Network & Distributed System Security Symposium, e Internet Society, San Diego, CA, USA (2013)
12. Porras, P., Cheung, S., Fong, M., Skinner, K., Yegneswaran, V.: Securing the software-defined network control layer. In: Proceedings of the 2015 Network and DistributedSystem Security Symposium (NDSS), San Diego, CA, USA, February 2015
13. Dhawan, M., Poddar, R., Mahajan, K., Mann, V.: Sphinx: detecting security attacks in software-defined networks. In Proceedings of the 2015 Network and Distributed System SecuritySymposium, vol. 15, pp. 8–11, San Diego, CA, USA, February 2015
14. Braga, R., Braga, E.M.M., Passito, A.: Lightweight ddos flooding attack detection using nox/openflow. In: Proceedings of the 2010 IEEE 35th Conference on LocalComputer Networks, ser. LCN '10, pp. 408–415, IEEE Computer Society, Washington, DC, USA, October 2010
15. Giotis, K., Androulidakis, G., Maglaris, V.: Leveraging sdn for efficient anomaly detection and mitigation on legacy networks. In: Proceedings of the 2014 'ird EuropeanWorkshop on Software Defined Networks, ser. EWSDN '14, pp. 85–90, IEEE Computer Society, Washington, DC, USA, September 2014
16. Chung, C.J., Khatkar, P., Xing, T., Lee, J., Huang, D.: NICE: network intrusion detection and countermeasure selection in virtual network systems. IEEE Trans. Dependable Secure Comput. **10**(4), 198–211 (2013)