International Journal of Security and Its Applications
Vol. 10, No. 2 (2016), pp.365-374
http://dx.doi.org/10.14257/ijsia.2016.10.2.32

An Extension Approach for Threat Detection and Defense of Software-Defined Networking

Hui Xu, Chunzhi Wang and Hongwei Chen

School of Computer Science, Hubei University of Technology, Wuhan, China
xuhui@mail.hbut.edu.cn

Abstract

Since the separation of the control plane and the data plane for Software-Defined Networking (SDN) leads to more and more security threats on the control plane and the SDN controllers, issues related to threat detection and defense of SDN must be seriously considered. This paper tries to introduce Extenics into the research on threat detection and defense of SDN and proposes an extension approach from a formal viewpoint. It first uses the matter-elements and the composite-elements to formally represent four security-related roles of SDN. It then utilizes the extension theory and basic-elements, by applying the dependent function to threat detection of SDN and making use of the affair-elements to formalize the NETCONF operations for threat defense of SDN. Finally, case study validates the feasibility of proposed extension approach in promoting the formalization of not only the security-related roles of SDN but also threat detection and defense of SDN.

Keywords*: Software-Defined Networking, threat detection and defense, basic-element, extension theory, dependent function*

1. Introduction

The separation of the control plane and the data plane for Software-Defined Networking (SDN) leads to more and more security threats on the control plane and the SDN controllers. Thus in this case, issues related to threat detection and defense of SDN must be seriously considered. And the OpenFlow Management and Configuration Protocol (OF-CONFIG, the newest version is 1.2 up to now) [1] proposed and developed by Open Networking Foundation (ONF) can then be utilized for study on threat detection and defense of SDN.

Reference [2] lists seven security threats for SDN, three of which are specific to SDN. Reference [3] focuses on two groups of Distributed Denial of Service (DDoS) attacks for SDN, one of which targets at the computing power and the other one of which exceeds available bandwidth, and discusses security vulnerabilities specific to SDN both on the data and control planes abused by attackers in DDoS attacks. Reference [4] proposes an approach to detect both IP spoofed packets and zombie PCs for SDN.

As for threat detection and defense of SDN, both common issues and SDN-specific issues must be taken into consideration. Thus, Extenics [5] prospects a promising way with the thinking of extension, and the logic cell of Extenics is basic-elements [6], which include matter-elements, affair-elements and relation-elements, and the formalization by their composition formats is called as composite-elements. Furthermore, the theory based on Extenics that is the extension theory might possibly be utilized for the research on threat detection and defense of SDN. The aim of this paper is then to propose an extension approach with the use of basic-elements and the extension theory from a formal viewpoint for threat detection and defense of SDN.

ISSN: 1738-9976 IJSIA

The remainder of this paper is organized as follows. Section 2 focuses on the formal representations for security-related roles of SDN using the matter-elements and the composite-elements. Section 3 applies the dependent function based on the extension theory to threat detection of SDN, and utilizes the affair-elements to formalize the NETCONF operations for threat defense of SDN. Section 4 discusses case study related to threat detection and defense of SDN and validates the feasibility of proposed extension approach. Section 5 concludes this paper.

2. Formal Representations for Security-related Roles of SDN

As the logic cell of Extenics, basic-elements are formally defined as the ordered three-tuple $B=(O,c,v)$, in which O are the objects with the characteristic c and the corresponding value v. And according to current version 1.2 of the OF-CONFIG protocol and its proposed data model based on YANG, this section tries to utilize the matter-elements and the composite-elements to formally describe four security-related roles of SDN in a unified manner, as a formal basis for threat detection and defense of SDN.

2.1. The Secure OpenFlow Configuration Point

The first main role to perform security detection and defense of SDN is the OpenFlow configuration point proposed by the OF-CONFIG protocol, formalized as the matter-element in Formula (1).

$$M_{scp}=\begin{bmatrix} O_{scp},id,v_{id} \\ url,v_{url} \\ protocal,v_{protocol} \\ traffic,v_{traffic} \end{bmatrix} \tag{1}$$

2.2. The Secure OpenFlow Controller

The second main role to perform security detection and defense of SDN is the OpenFlow controller, formalized as the matter-element in Formula (2).

$$M_{soc}=\begin{bmatrix} O_{soc},id,v_{id} \\ role,v_{role} \\ ip-address,v_{ia} \\ port,v_{port} \\ local-ip-address,v_{lia} \\ local-port,v_{lp} \\ protocal,v_{protocal} \\ state,M_{s} \\ request-count.v_{rc} \end{bmatrix} \tag{2}$$

As is shown in Formula (2), the OpenFlow controller relates to the state with its formalization as the matter-element in Formula (3).

$$M_s = \begin{bmatrix} O_s, connection-state, v_{cs} \\ current-version, v_{current-version} \end{bmatrix} \quad (3)$$

2.3. The Secure OpenFlow Logical Switch

According to current version 1.2 of the OF-CONFIG protocol, the OpenFlow controller controls the OpenFlow logical switch, which is formalized as the matter-element in Formula (4).

$$M_{sls} = \begin{bmatrix} O_{sls}, id, v_{id} \\ datapath-id, v_{did} \\ enabled, v_{enabled} \\ check-controller-certificate, v_{ccc} \\ lost-connection-behavior, v_{lcb} \\ reources, \{M_r\} \\ capabilities, \{M_{lsc}\} \\ controllers, \{M_{soc}\} \end{bmatrix} \quad (4)$$

As indicated in Formula (4), the OpenFlow logical switch not only relates to the OpenFlow controller, but also associates with the OpenFlow resource and the logical switch capability, respectively formalized as the matter-element in Formula (5) and the matter-element in Formula (6).

$$M_r = [O_r, resource-id, v_{rid}] \quad (5)$$

$$M_{lsc} = \begin{bmatrix} O_{lsc}, \max-buffered-packets, v_{mbp} \\ \max-tables, v_{mt} \\ \max-ports, v_{mp} \\ flow-statistics, v_{fs} \\ table-statistics, v_{ts} \\ port-statistics, v_{ps} \\ group-statistics, v_{gs} \\ queue-statistics, v_{qs} \\ reassemble-ip-fragments, v_{rif} \\ block-looping-ports, v_{blp} \\ reserved-port-types, \{M_t\} \\ group-types, \{M_t\} \\ group-capabilities, \{M_c\} \\ action-types, \{M_t\} \\ instruction-types, \{M_t\} \end{bmatrix} \quad (6)$$

As indicated in Formula (6), the logical switch capability associates with the type and the capability, respectively formalized as the matter-element in Formula (7) and the matter-element in Formula (8).

$$M_t = \left[O_t, type, v_{type}\right] \tag{7}$$

$$M_c = \left[O_c, capability, v_{capability}\right] \tag{8}$$

Note that, the OpenFlow logical switch can choose to check the certificate of the OpenFlow controller or not, which depends on its security strategy. There are two types of certificates, which are the external certificate and the owned certificate.

On one hand, the external certificate is introduced to be used by an OpenFlow logical switch for authenticating a controller when a TLS connection is established, which is formalized as the matter-element in Formula (9).

$$M_{ec} = \begin{bmatrix} O_{ec}, resource-id, v_{rid} \\ certificate, v_{certificate} \end{bmatrix} \tag{9}$$

On the other hand, the owned certificate is introduced to be used by an OpenFlow logical switch for authenticating itself to a controller when a TLS connection is established. Instances of an owned certificate contain a certificate and a private key, and the private key can be a DSA key value or a RSA key value. The owned certificate is then formalized as the matter-element in Formula (10).

$$M_{oc} = \begin{bmatrix} O_{oc}, resource-id, v_{rid} \\ certificate, v_{certificate} \\ key-value, v_{kv} \end{bmatrix} \tag{10}$$

2.4. The Cecure OpenFlow Capable Switc

As is shown in current version 1.2 of the OF-CONFIG protocol, the OpenFlow configuration point configures the OpenFlow capable switch, which is instantiated by the OpenFlow logical switch. Formula (11) formally defines the OpenFlow capable switch, which relates to the OpenFlow configuration point, the OpenFlow resource and the OpenFlow logical switch.

$$M_{scs} = \begin{bmatrix} O_{scs}, id, v_{id} \\ config-version, v_{config-version} \\ configuration-points, \{M_{scp}\} \\ resources, \{M_r\} \\ logical-switches, \{M_{sls}\} \end{bmatrix} \tag{11}$$

3. Applying the Extension Theory and Basic-elements to Threat Detection and Defense of SDN

Based on the formal representations for security-related roles of SDN using the matter-elements and the composite-elements, this section applies the extension theory and basic-elements to the research on threat detection and defense of SDN from a formal viewpoint.

3.1. Proposed Dependent Function for Threat Detection of SDN

As for threat detection of SDN, parameters such as computing power, available bandwidth and overall traffic should be monitored to find possible security threats. Both the acceptable value domain and the complete value domain of these parameters for threat detection of SDN can be defined in the real field.

Thus in this case, the extension theory is introduced and the dependent function is then proposed for threat detection of SDN. Note that, the extension theory extends the concept of distance to the concept of extension distance [7], which uses the negative value to indicate the distance between the point in a section and the section itself.

Definition 1 provides the extension distance between a detected value from monitoring a particular parameter for threat detection of SDN and a value domain that might be the acceptable value domain or the complete value domain of this parameter.

Definition 1 Suppose that $V = \langle x, y \rangle$ is a value domain of a particular parameter for threat detection of SDN in the real field, and v is a detected value from monitoring this parameter, then the extension distance between v and V can be defined as

$$d(v,V) = \left| v - \frac{x+y}{2} \right| - \frac{y-x}{2} \tag{12}$$

Definition 2 describes the extension distance between a detected value from monitoring a particular parameter for threat detection of SDN and two value domains that are the acceptable value domain and the complete value domain of this parameter.

Definition 2 Suppose that $V_s = \langle x_s, y_s \rangle$ and $V_f = \langle x_f, y_f \rangle$ are respectively the acceptable value domain and the complete value domain of a particular parameter for threat detection of SDN in the real field, $V_s \subseteq V_f$ and v is a detected value from monitoring this parameter, then the extension distance between v and these two related value domains V_s and V_f can be defined as

$$D(v,V_s,V_f) = d(v,V_f) - d(v,V_s) \tag{13}$$

As to the case of threat detection of SDN, the dependent function is proposed to confirm the association degree with the use of the acceptable value domain and the complete value domain, which is then formalized as Definition 3.

Definition 3 Suppose that $V_s = \langle x_s, y_s \rangle$ and $V_f = \langle x_f, y_f \rangle$ are respectively the acceptable value domain and the complete value domain of a particular parameter for threat detection of SDN in the real field, $V_s \subseteq V_f$ and v is a detected value from monitoring this parameter, then the dependent function between v and these two related domains V_s and V_f can be defined as

$$t(v)=\begin{cases}\dfrac{d(v,V_s)}{D(v,V_s,V_f)}, D(v,V_s,V_f)\neq 0, v\in V_f \\ -d(v,V_s)+1, D(v,V_s,V_f)=0, v\in V_s \\ 0, D(v,V_s,V_f)=0, v\notin V_s, v\in V_f\end{cases} \tag{14}$$

As is indicated in Definition 3, if $t(v)\geq 0$, it means that the detected value satisfies the acceptable value domain of the particular parameter for threat detection of SDN and it is currently secure for this parameter, and if $t(v)<0$, it means that the detected value is not related to the acceptable value domain of this parameter for threat detection of SDN and it reveals the security threat for SDN.

In summary, the proposed dependent function can be utilized to improve the formalization for threat detection of SDN from a quantitative point of view.

3.2. Affair-elements for Threat Defense of SDN using NETCONF Operations

Since current version 1.2 of the OF-CONFIG protocol adopts NETCONF [8] as its transport protocol, the NETCONF operations can then be utilized for threat defense of SDN. The basic NETCONF operations include <get-config>, <edit-config>, <copy-config> and <delete-config>, respectively with formalizations using the affair-elements as follows.

$$A_{get}=\begin{bmatrix}O_{get}, target, v_{target} \\ filter, v_{filter}\end{bmatrix} \tag{15}$$

$$A_{edit}=\begin{bmatrix}O_{edit}, target, v_{target} \\ default-operation, v_{do} \\ test-option, v_{to} \\ error-option, v_{eo} \\ config-object, \{v_{c-object}\} \\ config-operaion, \{v_{c-operation}\} \\ config-value, \{v_{c-value}\}\end{bmatrix} \tag{16}$$

$$A_{copy}=\begin{bmatrix}O_{copy}, target, v_{target} \\ source, v_{source}\end{bmatrix} \tag{17}$$

$$A_{delete}=\lfloor O_{delete}, target, v_{target}\rfloor \tag{18}$$

Take the NETCONF <edit-config> operation for example. It is formalized as the affair-element A_{edit}, which can be used to the managed instances for security-related roles of SDN by performing edit operations of some attributes from the viewpoint of threat defense.

4. Case Study

In order to validate the feasibility of the proposed extension approach for threat detection and defense of SDN, case study is discussed in this section.

4.1. Examples for Validation

First of all, when considering the formal representations for security-related roles of SDN, examples for a secure OpenFlow controller, a secure OpenFlow logical switch and a secure OpenFlow capable switch are respectively formalized as the following matter-elements, namely M_{soc1}, M_{sls1} and M_{scs1}.

$$M_{soc1} = \begin{bmatrix} O_{soc1}, id, openflow\_controller\_1 \\ role, master \\ ip-address, 192.168.6.1 \\ port, 6789 \\ protocal, tcp \end{bmatrix}$$

$$M_{sls1} = \begin{bmatrix} O_{sls1}, id, \log ical\_switch\_1 \\ datapath\text{–}id, 66:66:66:66:66:66:66:66 \\ enabled, true \\ controllers, \{M_{soc1}\} \end{bmatrix}$$

$$M_{scs1} = \begin{bmatrix} O_{scs1}, id, capable\_switch\_1 \\ \log ical-switches, \{M_{sls1}\} \end{bmatrix}$$

The action of creating a capable-switch configuration with the use of a NETCONF <edit-config> operation is then formalized as the following affair-element.

$$A_{edit1} = \begin{bmatrix} O_{edit1}, target, candidate \\ default\text{–}operation, merge \\ test-option, set \\ config-object, \{M_{scs1}\} \\ config-operaion, \{create\} \end{bmatrix}$$

4.2. Analysis with the use of Proposed Extension Approach

When the secure OpenFlow controller in the examples above suffers from a DDoS attack in SDN, assume that, a) the selected parameter for threat detection of SDN is the utilization ratio for the secure OpenFlow controller, b) the acceptable value domain and the complete value domain of this parameter for threat detection of SDN are respectively $V_s = [0,0.8]$ and $V_f = [0,1]$, and c) current detected value of the utilization ratio for the secure OpenFlow controller is $v = 0.9$.

Firstly, according to Formula (12), $d(v,V_s)=\left|0.9-\frac{0+0.8}{2}\right|-\frac{0.8-0}{2}=0.1$ and $d(v,V_f)=\left|0.9-\frac{0+1}{2}\right|-\frac{1-0}{2}=-0.1$.

Then, according to Formula (13), $D(v,V_s,V_f)=d(v,V_f)-d(v,V_s)=-0.1-0.1=-0.2$.

Furthermore, according to Formula (14), since there is $D(v,V_s,V_f)\neq 0, v\in V_f$, $t(v)=\frac{d(v,V_s)}{D(v,V_s,V_f)}=\frac{0.1}{-0.2}=-\frac{1}{2}<0$, which reveals that current detected value of the utilization ratio for the secure OpenFlow controller is not related to the acceptable value domain of this parameter, and the security threat might possibly exist for SDN.

Thus, a possible defense policy is to replace the ip-address of the secure OpenFlow controller with the use of the NETCONF <edit-config> operation formalized as the following affair-element A_{edit2}.

$$A_{edit2}=\begin{bmatrix} O_{edit2}, target, candidate \\ default-operation, merge \\ config-object, \{M_{soc1}: ip-address\} \\ config-operaion, \{replace\} \\ config-value, \{192.168.32.1\} \end{bmatrix}$$

When A_{edit2} is performed to M_{soc1}, the secure OpenFlow controller replaces its ip-address and it is then formalized as M'_{soc1}.

$$M_{soc1}\xrightarrow{A_{edit2}}\begin{bmatrix} O_{soc1}, id, openflow\_controller\_1 \\ role, master \\ ip-address, 192.168.32.1 \\ port, 6789 \\ protocal, tcp \end{bmatrix}=M'_{soc1}$$

Case study shows that, proposed extension approach utilizes the basic-elements and the dependent function, in order to promote the formalization of not only the security-related roles of SDN but also the threat detection and defense for SDN.

5. Conclusions

This paper focuses on issues related to threat detection and defense of SDN, and the main contribution of this paper is to propose an extension approach based on Extenics from a formal viewpoint. This paper first uses the matter-elements and the composite-elements to formally represent four security-related roles of SDN, including the secure OpenFlow configuration point, the secure OpenFlow controller, the secure OpenFlow logical switch and the secure OpenFlow capable switch. It then utilizes the extension theory by applying the dependent function to threat detection of SDN and makes use of the affair-elements to formalize the NETCONF operations for threat defense of SDN. Finally, it validates the feasibility of proposed extension approach by case study.

Acknowledgements

This work has been supported by the Emergency Management Program for National Natural Science Foundation of China (No. 61440024), the Provincial Teaching Reform Research Project of Education Department of Hubei Province in China (No. 2012273), the Doctoral Scientific Research Fund from Hubei University of Technology (No. BSQD12029), the National Natural Science Foundation of China for Young Scholars (No. 61202287, No. 41301371) and the General Program for National Natural Science Foundation of China (No. 61170135). The authors would like to thank all project partners for their valuable contributions and feedbacks.

References

[1] Open Networking Foundation. OpenFlow Management and Configuration Protocol 1.2 (OF-CONFIG 1.2). Available: www.opennetworking.org/technical-communities/areas/specification/ 1928-of-config **(2015)**
[2] D. Kreutz, F. M. V. Ramos and P. Verissimo. Towards Secure and Dependable Software-Defined Networks. Proceeding of 2nd ACM SIGCOMM Workshop on Hot Topics in Software Defined Networking **(2013)**, pp. 55-60
[3] M. Vizváry and J. Vykopal. Future of DDoS Attacks Mitigation in Software Defined Networks. Lecture Notes in Computer Science, Vol. 8508 **(2014)**, pp. 123-127
[4] H. B. Bae, M. W. Park and S. H. Kim et al. Zombie PC Detection and Treatment Model on Software-Defined Network. Lecture Notes in Electrical Engineering, Vol. 330 **(2015)**, pp. 837-843
[5] C. Y. Yang and W. Cai. Extenics. Science Press, Beijing **(2014)**, Simplified Chinese version
[6] W. Cai, C. Y. Yang and B. He. Preliminary Extension Logic. Science Press, Beijing **(2003)**, Simplified Chinese version
[7] C. Y. Yang and W. Cai. Recent Research Progress in Dependent Functions in Extension Sets. Journal of Guangdong University of Technology, Vol. 29 **(2012)**, No. 2, pp. 7-14, in Chinese
[8] R. Enns, M. Bjorklund, J. Schoenwaelder and A. Bierman eds. Network Configuration Protocol (NETCONF). RFC6241 **(2011)**

Authors

Hui Xu, She received a bachelor's degree in Computer Science and Technology from Huazhong Normal University, Wuhan, China in 2005, a master's degree in Computer Application Technology from Huazhong Normal University, Wuhan, China in 2008, and a doctor's degree in Radio Physics from Huazhong Normal University, Wuhan, China in 2010. Since 2006, she has been a certified computer system analyst in China. Now, she is an Associate Professor at the School of Computer Science in Hubei University of Technology, Wuhan, China. Currently, her major field of study is network and service management.

Dr. Xu became a Member of Institute of Electrical and Electronics Engineers (IEEE) in 2007, a Member of Association for Computing Machinery (ACM) in 2007 and a Member of China Computer Federation (CCF) in 2008. She has authored or coauthored 1 book and 2 book chapters in the field of network management, about 10 papers published by Chinese journals, more than 10 papers published by international journals, and more than 20 papers published by international conferences. In April 2008, she was awarded by International Association of Engineers (IAENG) for her first-authored paper presented to 2008 IAENG International Conference on Communication Systems and Applications. Additionally, she was a Session Co-Chair or a Paper Reviewer for 2nd&3rd&7th&8th

International Conference on Computer Science and Education (ICCSE 2007&2008&2012&2013), a Session Chair for 1st International Symposium on Electronic Commerce and Security (ISECS 2008), a Paper Reviewer for 4th IEEE Conference on Industrial Electronics and Applications (ICIEA 2009), a Paper Reviewer for 3rd International Conference on Computer and Network Technology (ICCNT 2011), a Paper Reviewer for 32nd Chinese Control Conference (CCC 2013), and a Paper Reviewer for Security and Communication Networks, an international journal published by Wiley Press.

Chunzhi Wang, She is a Professor at the School of Computer Science in Hubei University of Technology, Wuhan, China. She is also the Dean of the School of Computer Science in Hubei University of Technology, Wuhan, China. Currently, her major field of study is Software-Defined Networking.

Hongwei Chen, He is a Professor at the School of Computer Science in Hubei University of Technology, Wuhan, China. Currently, his major field of study is distributed management.