Draft Recommendation ITU-T X.sdnsec-1

Security services using the software-defined network

CONTENTS

Draft Recommendation ITU-T X.sdnsec-1

Security services using the software-defined network

Summary

This recommendation is to support the protection of network resources using security services based on software-defined networking (SDN). First of all, this classifies the network resource for SDN-based security services. These network resources are SDN application, SDN controller, SDN switch and security manager (SM). And then this defines security services based on SDN.

Keywords

Access control, DDoS attack, firewall, honeypot, Software-Defined Networking (SDN), Security Scenarios

Introduction

Due to the increase of sophisticated network attacks, the legacy security services become difficult to cope with such network attacks in an autonomous manner. Software-defined networking (SDN) has been introduced to make networks more controllable and manageable, and this SDN technology will be promising to autonomously deal with such network attacks in a prompt manner.

This Recommendation describes valuable security services and scenarios for intra- and/or inter-domain. Also, this raises the specific requirements in each scenario.

For the centralized firewall service related to the intra-domain services, this Recommendation raises limitations in legacy firewalls in terms of flexibility and administration costs. Since in many cases, access control management for firewall is manually performed, it is difficult to add the access control policy rules corresponding to new network attacks in a prompt and autonomous manner. Thus, this situation requires expensive administration costs. This Recommendation introduces a SDN-based firewall service to overcome these limitations.

For the centralized distributed denial-of-service-attack (DDoS-attack) mitigation service related to the inter-domain services, this Recommendation raises limitations in legacy DDoS-attack mitigation techniques in terms of flexibility and administration costs. Since in many cases, network configuration for the mitigation is manually performed, it is difficult to dynamically configure network devices to limit and control suspicious network traffic for DDoS attacks. This Recommendation introduces a SDN-based DDoS-attack mitigation service to provide an autonomous and prompt configuration for suspicious network traffic.

For the centralized honeypot service related to the host devices in the intra-domain, this Recommendation raises limitations in legacy honeypots in terms of flexibility and administration costs. Since in many cases, network configuration for the honeypot is manually performed, it is difficult to dynamically configure honeypots to monitor and respond to attacks in real time. This Recommendation introduces a SDN-based honeypot service to provide an autonomous and prompt configuration for network-based honeypot service.

For the centralized illegal device management service related to the inter-domain services, this Recommendation raises limitations in localized illegal device management services in terms of scalability and compatibility. Since in many cases, illegal device management is locally performed, it is difficult to globally maintain the blacklists of illegal devices and/or hosts to prevent the traffic from those devices/hosts. This Recommendation introduces an illegal device management service to provide an autonomous and prompt configuration for illegal device management.

1 Scope

This Recommendation is to support the protection of network resources using security services based on software-defined networking (SDN). This recommendation covers as follows:

– Classify the network resources for SDN-based security services;

– Define security services based on SDN;

– Specify how to implement SDN-based security services

The protection of network resources (e.g., router, switch, firewall and intrusion detection system (IDS)) in security services based on SDN means:

– Prompt reaction to new network attacks (e.g., worms and DDoS attacks);

– Construction of private networks to mitigate sophisticated network attacks;

– Automatic defense from network attacks without the intervention of network administrators;

– Dynamic network-load-aware resource allocation.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T Y.3001] Recommendation ITU-T Y.3001 (2011), *Future Networks: Objectives and Design Goals*

[ITU-T Y.3011] Recommendation ITU-T Y.3011 (2012), *Framework of network virtualization for Future Networks*

[ITU-T Y.3300] Recommendation ITU-T Y.3300 (2014), *Framework of software-defined networking*

[ITU-T Y.3301] Recommendation ITU-T Y.3301 (2016), *Functional requirements of software-defined networking*

[ITU-T Y.3302] Recommendation ITU-T Y.3302 (2016), *Functional architecture of software-defined networking*

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 software-defined networking [ITU-T Y.3300]: A set of techniques that enables to directly program, orchestrate, control and manage network resources, which facilitates the design, delivery and operation of network services in a dynamic and scalable manner.

3.1.2 access control [b-ITU-T X.1252]: A procedure used to determine if an entity should be granted access to resources, facilities, services, or information based on pre-established rules and specific rights or authority associated with the requesting party.

3.1.3 access control policy [b-ITU-T X.800]: The set of rules that define the conditions under which and access may take place.

3.1.4 access control policy rules [b-ITU-T X.800]: Security policy rules concerning the provision of the access control service.

3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

3.2.1 network resources: Network devices that can perform packet forwarding in a network system. The network resources include network switch, router, gateway, WiFi access points, and similar devices.

3.2.2 firewall: A firewall that is a device or service at the junction of two network segments that inspects every packet that attempts to cross the boundary. It also rejects any packet that does not satisfy certain criteria for disallowed port numbers or IP addresses.

3.2.3 honeypot: A honeypot that is a device or service ready to be attacked for collecting attack data. The term honeypot comes from its behavior, which attracts attackers (bees) to a place (the attack target, or “honey”) used as a trap.

3.2.4 centralized firewall service: A centralized firewall that can establish and distribute access control policy rules into network resources for the efficient firewall management. These rules can be managed dynamically by a centralized server. SDN can work as a network-based firewall system through a standard interface between firewall applications and network resources.

3.2.5 centralized DDoS-attack mitigation service: A centralized mitigator that can establish and distribute access control policy rules into network resources for the efficient DDoS-attack mitigation. These rules can be managed dynamically by a centralized server. SDN can work as a network-based mitigation system through a standard interface between DDoS-attack mitigation applications and network resources.

3.2.6 centralized honeypot service: A centralized honeypot that can establish and distribute access control policy rules into network resources for the dynamic honeypot configuration. These rules can be managed dynamically by a centralized server. SDN can work as a network-based honeypot system through a standard interface between honeypot applications and network resources.

3.2.7 centralized illegal device management service: A centralized illegal device manager that can establish and distribute access control policy rules into network resources for the blacklist of illegal devices. These rules can be managed dynamically and globally by a centralized server. SDN can work as a network-based illegal device management system through a standard interface between illegal device management applications and network resources.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

ACI	Application Control Interface
ACM	Access Control Management
AL	Application Layer
AL-MSO	Application Layer Management Support and Orchestration
ALM	Application Layer Management
BSS	Business Support System
CL	Control Layer
CL-AS	Control Layer Application Support
CL-CLS	Control Layer Control Layer Services
CL-MSO	Control Layer Management Support and Orchestration
CL-RA	Control Layer Resource Abstraction
CLM	Control Layer Management
DDoS	Distributed Denial-of-Service
DNS	Domain Name Services
IDS	Intrusion Detection System
MMF	Multi-layer Management Function
MMFA	Multi-layer Management Function Application layer
MMFC	Multi-layer Management Function Control layer
MMFO	Multi-layer Management Function
MMFR	Multi-layer Management Function Resource layer
NBI	Northbound Interface
OSS	Operation Support System
RCI	Resource Control Interface
RL	Resource Layer
RL-MS	Resource Layer Management Support
RLM	Resource Layer Management
SBI	Southbound Interface
SDN	Software-Defined Networking
SDN-AL	Software-Defined Networking - Application Layer
SDN-CL	Software-Defined Networking - Control Layer
SDN-RL	Software-Defined Networking - Resource Layer
SM	Security Manager
TCP	Transmission Control Protocol

5 Conventions

In this Recommendation:

The keywords "**is required to**" indicate a requirement which must be strictly followed and from which no deviation is permitted, if conformance to this Recommendation is to be claimed.

The keywords "**is recommended**" indicate a requirement which is recommended but which is not absolutely required. Thus, this requirement need not be present to claim conformance.

The keywords "**is prohibited from**" indicate a requirement which must be strictly followed and from which no deviation is permitted, if conformance to this Recommendation is to be claimed.

The keywords "**can optionally**" indicate an optional requirement which is permissible, without implying any sense of being recommended. This term is not intended to imply that the vendor's implementation must provide the option, and the feature can be optionally enabled by the network operator/service provider. Rather, it means the vendor may optionally provide the feature and still claim conformance with the specification.

6 Overview of the SDN functional architecture

This clause describes the high-level reference architecture to support SDN-based security services, such as centralized firewall system and centralized DDoS-attack mitigation system.

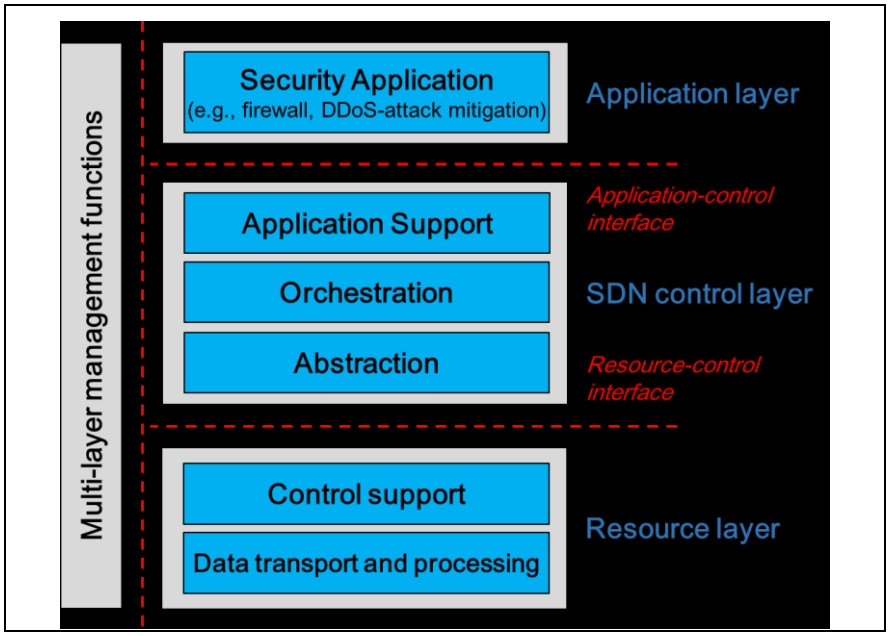


Figure 6-1 – High-level architecture for SDN-based security services (ITU-T Y.3300)

As shown in Figure 6-1, applications for security services (e.g., firewall, DDoS-attack mitigation and honeypot applications) run on the top of SDN

DN controller. When an administrator enforces security policies for the security services through an application interface, SDN controller generates the corresponding access control policy rules to meet such security policies in an autonomous and prompt manner. According to the generated access control policy rules, the network resources such as switches take an action to mitigate network attacks, for example, dropping packets with suspicious patterns.

Figure 6-2 shows the functional architecture of SDN, which is based on the high-level architecture of SDN of Figure 6-1.

- SDN Application layer (SDN-AL): SDN-AL consists of Application Layer Management Support and Orchestration (AL-MSO) and many SDN applications. The AL-MSO interacts with Application Layer Management (ALM) functional component in Multi-layer Management Function (MMF) via Multi-layer Management Functions Application layer (MMFA) reference point in order to support management of SDN applications and to enable joint-operations of management in all SDN sub-layers. SDN applications specify how network resources should be controlled by interacting with the SDN control layer (SDN-CL) via Application Control Interfaces (ACIs). Therefore, the SDN-AL uses the abstracted view and status of the network resources which are provided by the SDN-CL by means of information and data models exposed via ACI. Depending on the SDN use case (e.g. intra or inter data centers, mobile networks, access networks) different ACI can be defined. It is assumed that ACI will use Open APIs.

– SDN Control Layer (SDN-CL): SDN-CL consists of Control Layer Management Support and Orchestration (CL-MSO), Application Support (CL-AS), Control Layer Services (CL-CLS) and Resource Abstraction (CL-RA). the SDN-CL provides a programmable means to control the behavior of SDN resources (such as data transport and processing resources), according to SDN-AL requests and MMF policies. The SDN-CL is operating on resources provided by the SDN Resource Layer (SDN-RL) and exposes an abstracted view of the network to the SDN-AL. The SDN-CL interacts with SDN-RL using Resource Control Interfaces (RCI) reference point, with control layer management (CLM) functional component in MMF using Multi-layer Management Function Control layer (MMFC) reference point. It also interacts with SDN-AL with ACI reference point. The CL-MSO of SDN-CL requests to MMF in order to delegate some management functions which are then realized by MMF functional components. MMF is responsible for the management of SDN-CL through the MMFC reference point.

– SDN Resource Layer (SDN-RL): SDN-RL consists of Resource Layer Management Support (RL-MS), Resource Layer Control Support (RL-CS), Resource Layer Data Processing (RL-DP), Resource Layer Data Transport (RL-DT) and Resource Layer Management Support(RL-MS). SDN-RL is where the physical or virtual network elements perform transport and/or processing of data packets according to SDN-CL decisions. These decisions as well as the information about network resources are exchanged via Resource Control Interface (RCI) reference point. The Resource Layer Management Support (RL-MS) of SDN-RL also interacts with MMF using Multi-layer Management Function Resource layer (MMFR) reference point. Information exchanged through RCI include control information provided by SDN-CL to SDN-RL (e.g., for configuring a network resource or providing policies) as well as the information that pertains to the (unsolicited) notifications sent by SDN-RL whenever a network resource change is detected (if such information is available).

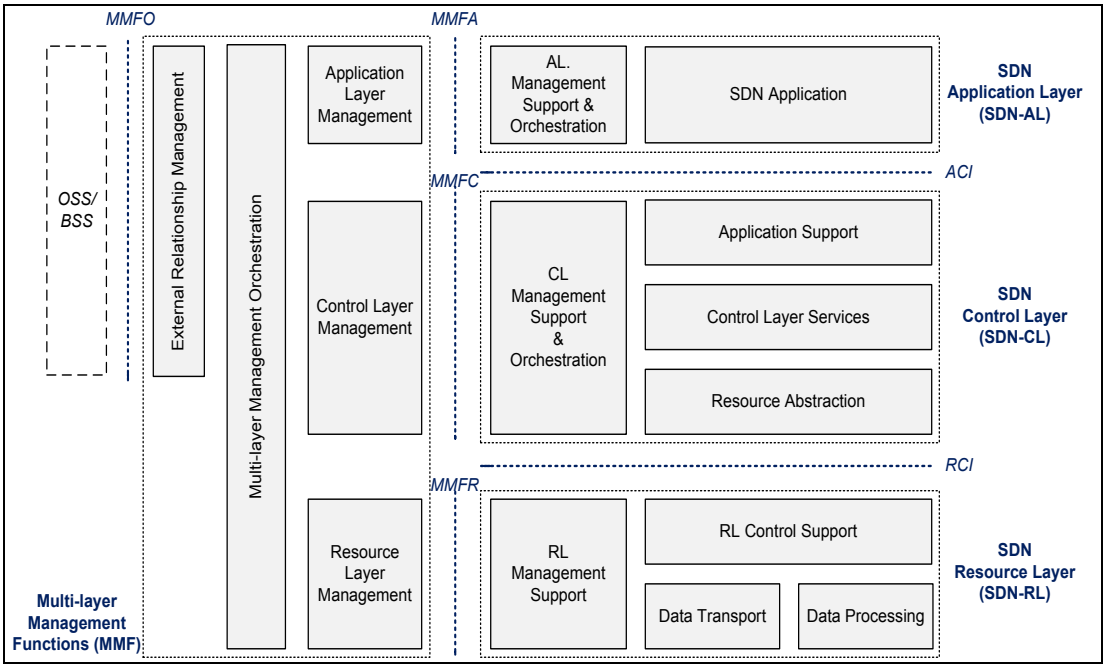


Figure 6-2 – SDN functional architecture [ITU-T Y.3302]

7 Classification of network resources

This clause introduces network resources for security services using SDN. According to the Figure 6-2, this clause classifies network resources into three types: SDN application, SDN controller and SDN switch.

- SDN application is a program that explicitly, directly, and programmatically communicate their network requirements and desired network behavior to SDN Controller via a northbound interface (NBI) such as the ACL in Figure 6-2. In addition, they may consume an abstracted view of the network for their internal decision making purposes. For example, firewall, honeypot, DDoS mitigation and illegal device management services can be provided as applications.
- SDN Controller is a logically centralized entity in charge of (i) translating the requirements from applications to SDN switches and (ii) providing abstract network views to applications with useful network information such as traffic statistics and events.
- SDN switch is a software program or hardware device that forwards packets in a SDN environment. SDN switches are capable of storing packet forwarding rules managed by a SDN controller via a southbound interface (SBI) such as RCL in Figure 6-2.

Network resource, security manager (SM), is also defined. The SM transfers policy rules to SDN application. Figure 7-1 shows the location of network resources in SDN functional architecture.

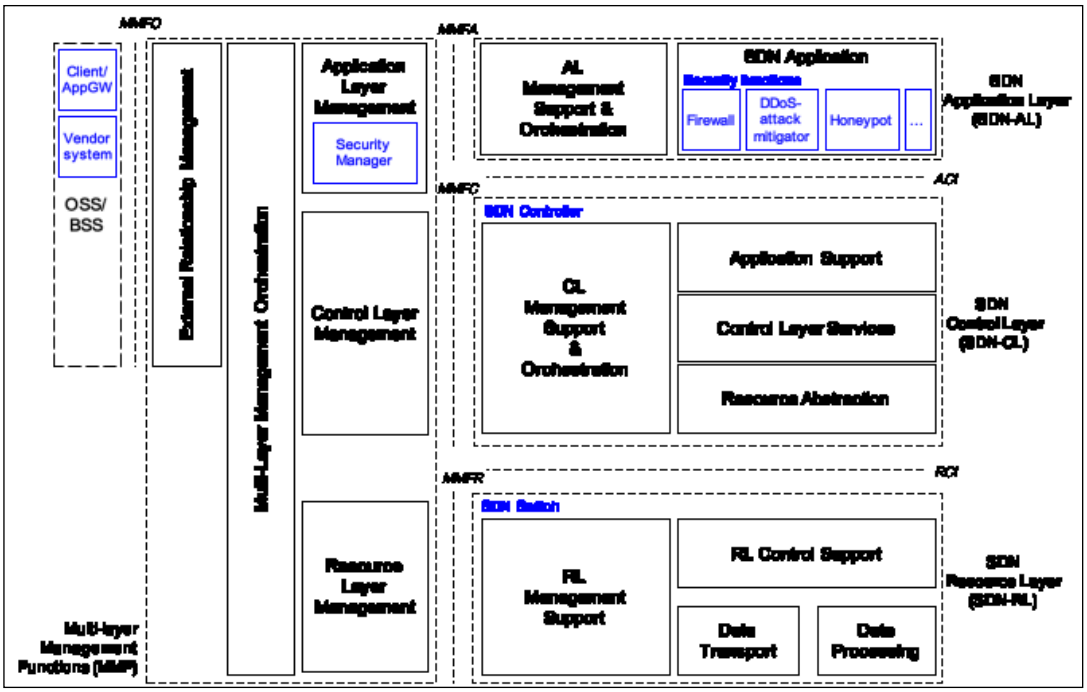


Figure 7-1 – Network resources in SDN-based security services

8 Security services based on SDN

This clause introduces security services using SDN in two kinds of networks: (i) Intra-domain network (e.g., centralized firewall service and centralized honeypot service) and (ii) Inter-domain networks (e.g., centralized DDoS-attack mitigation service and centralized illegal device management service).

8.1 Centralized firewall service

8.1.1 Basic concept of centralized firewall service

This clause describes the basic concept of centralized firewall service. This service can manage network resources and firewall rules can be managed flexibly. As shown in Figure 8-1, a centralized firewall manages SDN switches and firewall rules can be added into them or deleted.

Note：It is easy to convert a packet-filtering strategy, which is issued by the firewall application, to a flow table through the controller. However, a protocol between controller and switches (e.g., openflow and netconf) is currently only able to match up to the TCP layer, and there is no corresponding field to set the identification information of data packet above TCP layer. So it can't be achieved to identify the information above the TCP layer firewall strategy without changing the protocol.

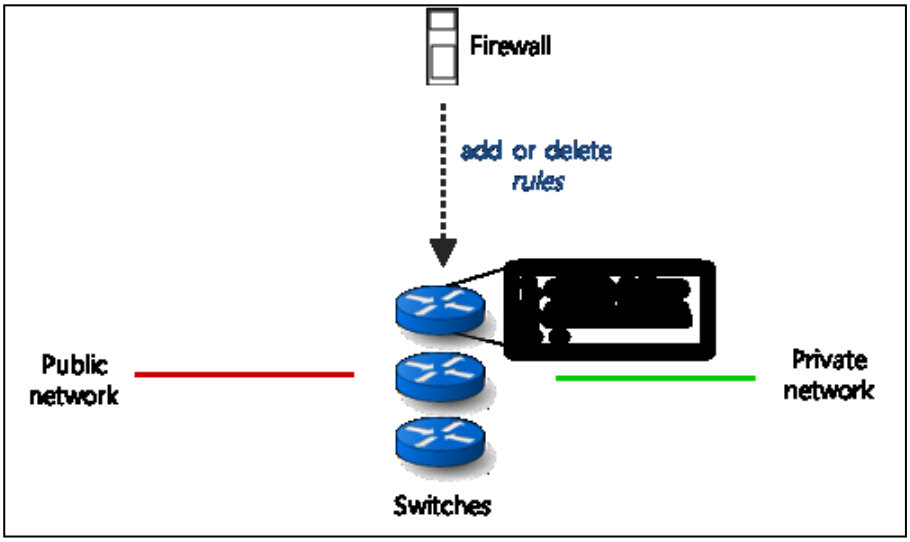


Figure 8-1 – Centralized firewall service in intra-domain

8.1.2 Service scenario of centralized firewall service

Figure 8-2 shows an example scenario of centralized firewall service for switches. This clause describes processes of stopping worm spreading. This scenario shows that how a user can manage a centralized firewall service. This scenario concentrates on SDN switches.

As a precondition for this scenario, a SM should specify a new policy to firewall application when the information about a new worm is recognized. In order to prevent packets from including this worm, the user adds the new policy (e.g., "Drop packets with the worm file") to the firewall application running on top of the SDN controller. It can also be managed centrally such that a SM might determine security policies for firewall application through a single point, that is, SDN controller.

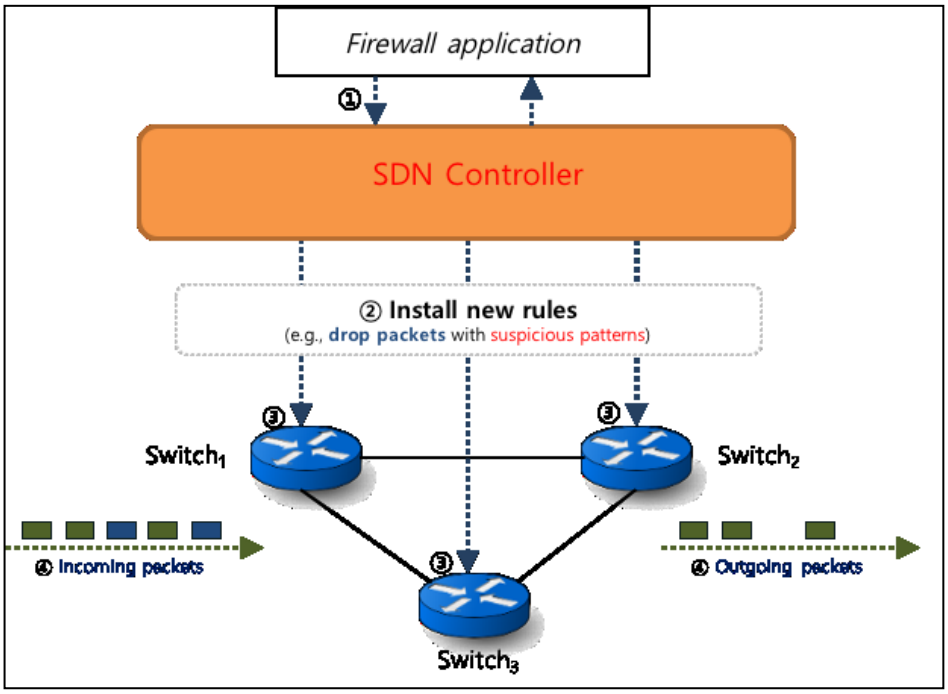


Figure 8-2 – Example scenario of centralized firewall service

- [Step 1] A firewall application installs new rules

A firewall application should specify a new rule when the information about a new worm is reported. The new rule (e.g., "Drop packets with the worm file") is added to SDN controller.

- [Step 2] The SDN controller distributes a new flow entry to all SDN switches

A new **flow entry** might be distributed to each switch by a SDN controller after installing it. Therefore, the SDN controller sends a flow insert operation that contains the rule (e.g., "Drop packets with the worm file") to all SDN switches.

The reported new worm described as above is either a known worm or a "zero-day" worm. As for a known worm, some mechanisms such as "signatures" and "thumbprints" are developed in firewall service to detect and defend it. However, for a "zero-day" worm, it should be scanned and detected before any countermeasure is applied to defend it. Worms deliver malicious payloads that could exploit some vulnerable applications or services. Those worms might be detected by inspecting the packet payload. An example of packet data scan detection is shown in Appendix II.

- [Step 3] All SDN switches apply to new flow entry in their flow table

An SDN switch adds a flow entry dropping future packets with the worm file to its flow table when receiving the flow insert operation about the worm file. After that, the SDN switch can drop the packets with the worm file.

- [Step 4] The SDN switch executes flow entries to drop packets including worm file

An SDN switch completely drops packets when receiving packets with worm file. Any packets with worm file cannot be passed switch under applied rules.

- [Step 5] An SDN switch reports to controller when receiving an unfamiliar packet.

When an SDN switch receives a type of packet that it never processed before, it deletes this packet and sends a report to the controller about this kind of packets. The controller analyzes whether this is an attack. If this is an attack, controller sends a message to the firewall application and Step 1 will be executed. If not, the controller keeps a regular flow entry to tell switches how to handle this sequence of afterwards packet.

8.2 Centralized honeypot service

8.2.1 Basic concept of centralized honeypot service

This clause describes the basic concept of centralized honeypot service. The centralized honeypot can manage honeypot places. As shown in Figure 8-3, a centralized honeypot manages switches and new routing paths to the honeypots to attract attackers to a place used as a trap. The honeypot is configured as the intended attack target and reports the collected information to the centralized honeypot service.

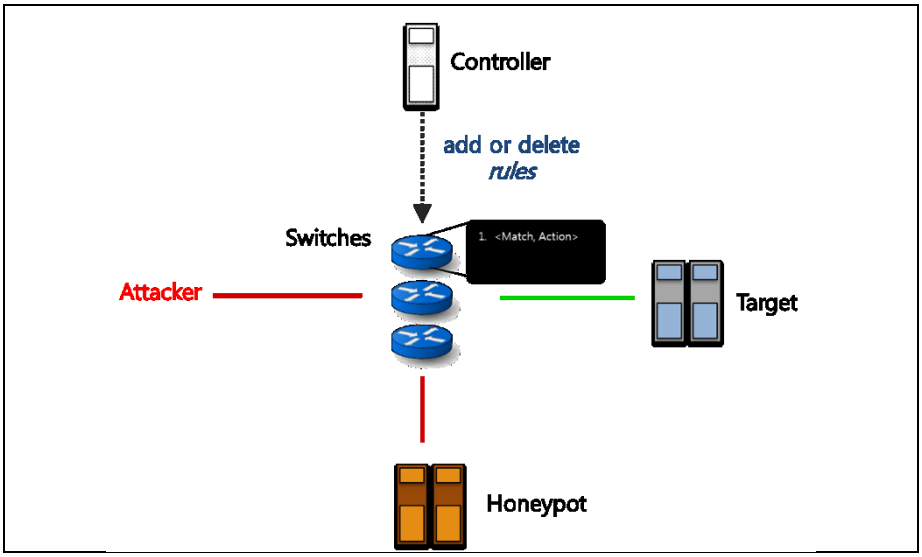


Figure 8-3 – Centralized honeypot service in intra-domain

8.2.2 Service scenario of centralized honeypot

Figure 8-4 shows a centralized honeypot service for switches. This clause defines processes of adding a routing path to a honeypot instead of the actual target. Adding a routing path to a honeypot scenario shows that how a SM can use a centralized honeypot system. This scenario concentrates on SDN switches.

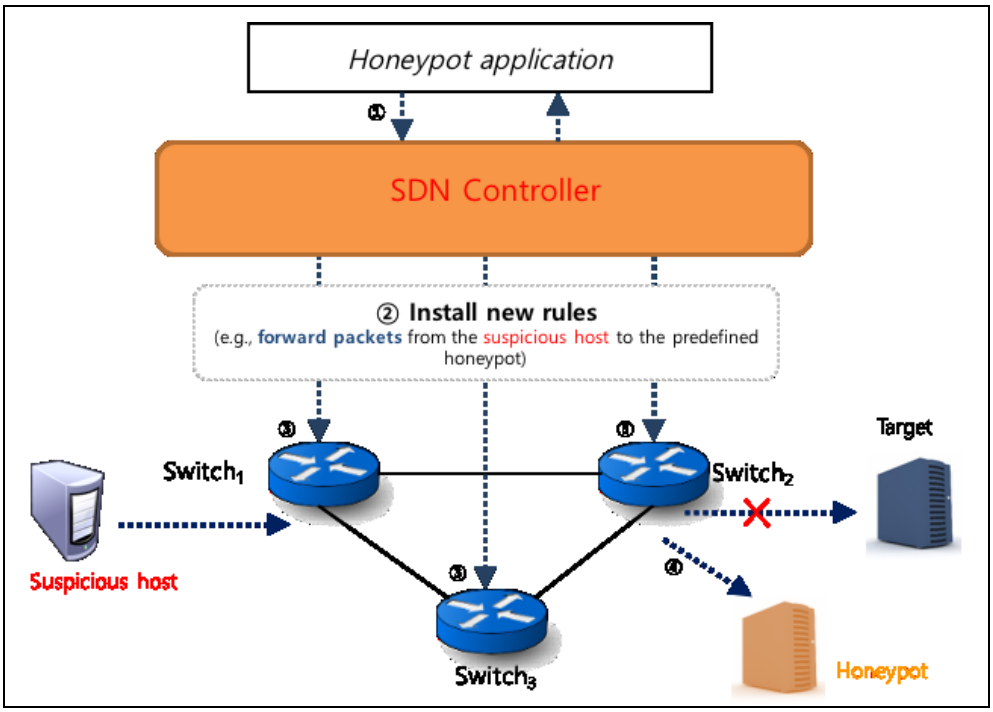


Figure 8-4 –Example scenario for centralized honeypot service

- [Step 1] A honeypot application installs new rules to SDN controller

A honeypot application should specify a new rule when the information about some suspicious host is reported. In order to monitor the traffic from suspicious host, the new rule (e.g., “Forward packets from the suspicious host to a honeypot”) is added to SDN controller by honeypot application running on top of the SDN controller.

- [Step 2] A SDN controller distributes new rules to appropriate SDN switches

A new rule might be distributed to each switch by an SDN controller after installing it. Therefore, the SDN controller sends a flow insert operation that contains the rule (e.g., “Forward packets from the suspicious host to a honeypot”) to all SDN switches. It can also be managed centrally such that a SM can determine security policies for their service through a single point, that is, SDN controller.

- [Step 3] All SDN switches apply to new rules in their flow table.

All SDN switches add a flow entry forwarding future packets from the suspicious host to a honeypot to their flow tables when receiving the flow insert operation about the suspicious host. After that, the SDN switch can forward the packets from the suspicious host to a honeypot.

- [Step 4] A SDN switch executes new rules to support honeypot service

An SDN switch SDN switch can forward the packets to a honeypot when receiving packets from the suspicious host. Any packets from the suspicious host cannot be passed to an actual target host switch under applied rules. The forwarded packets are collected in the honeypot.

8.3 Centralized DDoS-attack mitigation service

8.3.1 Basic concept of centralized DDoS-attack mitigation service

Figure 8-5 shows a centralized DDoS-attack mitigation service. This service adds, deletes or modifies rules to each switch. Unlike the “Centralized firewall service” related to the intranet services, this service mainly focuses on the inter-domain level.

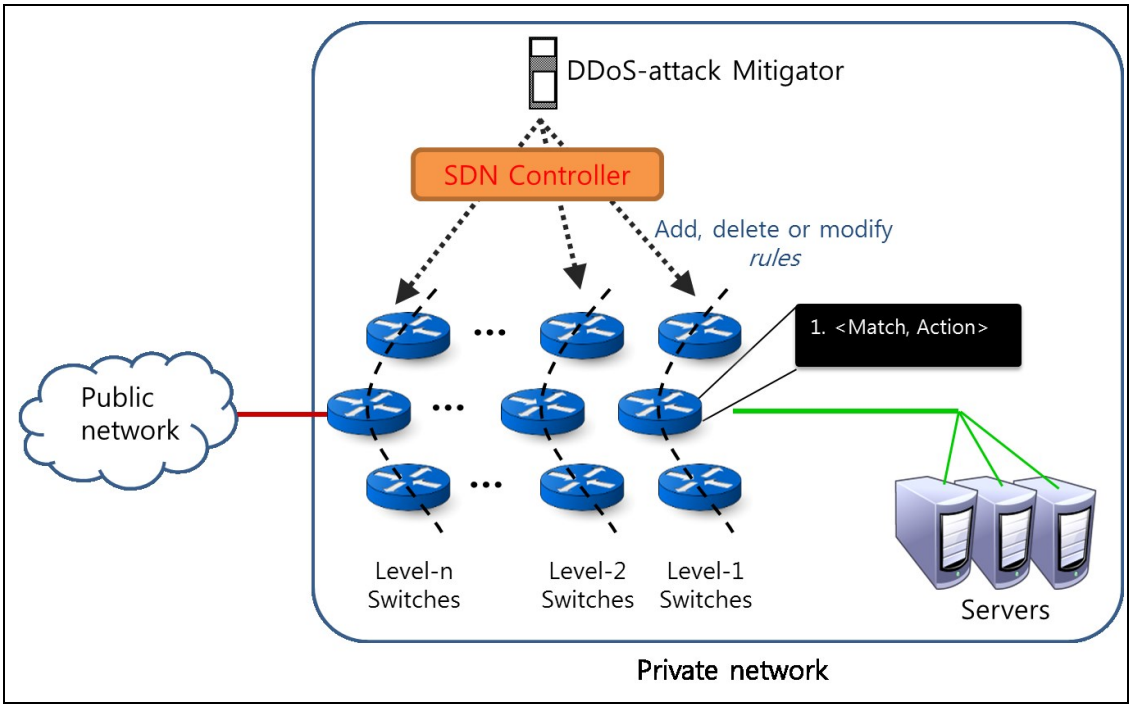


Figure 8-5 – Centralized DDoS-attack mitigation service in inter-domain

8.3.2 Services scenarios of centralized DDoS-attack mitigation service

8.3.2.1 A service scenario of centralized DDoS-attack mitigation for stateless servers

Figure 8-6 shows an example scenario of centralized DDoS-attack mitigation for stateless servers. This clause defines processes against Domain Name Services (DNS) DDoS attacks. In Figure 8-6, mitigating DDoS attacks for stateless server's scenario shows that how SM can manage a centralized DDoS-attack mitigation. This scenario concentrates on SDN switches.

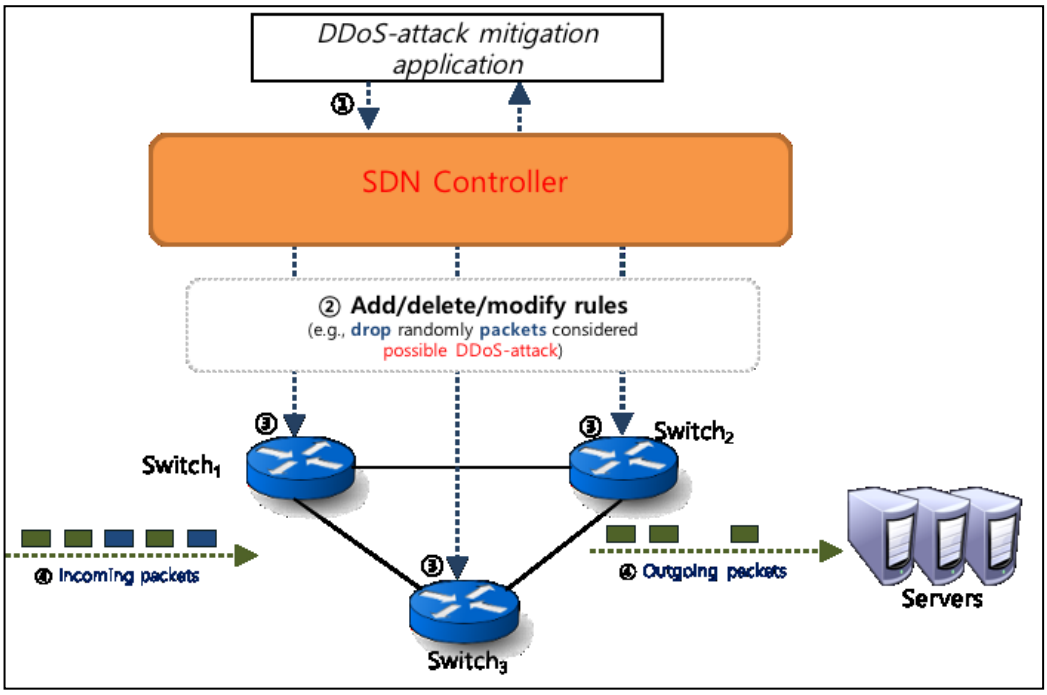


Figure 8-6 – Example scenario for centralized DDoS-attack mitigation for stateless servers

- [Step 1] A mitigator application installs new rules to SDN controller

A DDoS-attack mitigator application should specify a new rule when a new DDoS-attack is detected. In order to prevent packets from reaching servers for wasting servers' resources, the new rule (e.g., "Drop DDoS-attack packets randomly with some probability") is added to SDN controller. This rule addition is performed by DDoS-attack mitigation application running on top of the SDN controller.

- [Step 2] A SDN controller distributes new rules to appropriate switches

A new rule might be distributed to each switch by a SDN controller after installing it. Therefore, the SDN controller sends a flow insert operation that contains the rule (e.g., "Drop randomly packets considered DDoS attacks with a certain probability") to all SDN switches. It can also be

managed centrally such that a SM can determine security policies for their service through a single point, that is, SDN controller.

- [Step 3] All SDN switches apply to new rules in their flow tables

All SDN switches add a flow entry dropping future packets considered the DDoS-attack packets to their flow tables when receiving the flow insert operation about the DDoS-attack mitigation. After that, the SDN switch can drop the DDoS-attack packets with a probability proportional to the severity of the DDoS-attack.

- [Step 4] A SDN switch executes new rules to mitigate DDoS-attack

An SDN switch completely drops packets selectively when receiving DDoS-attack packets. DDoS-attack packets are dropped randomly through multi-level switches.

8.3.2.2 A service scenario of centralized DDoS-attack mitigation for stateful servers

Figure 8-7 shows a centralized DDoS-attack mitigation for stateful servers. This clause defines processes against Web-server DDoS attacks. In Figure 8-7, mitigating DDoS attacks for stateful server scenario shows that how a SM can manage a centralized DDoS-attack mitigation. This scenario concentrates on SDN switches.

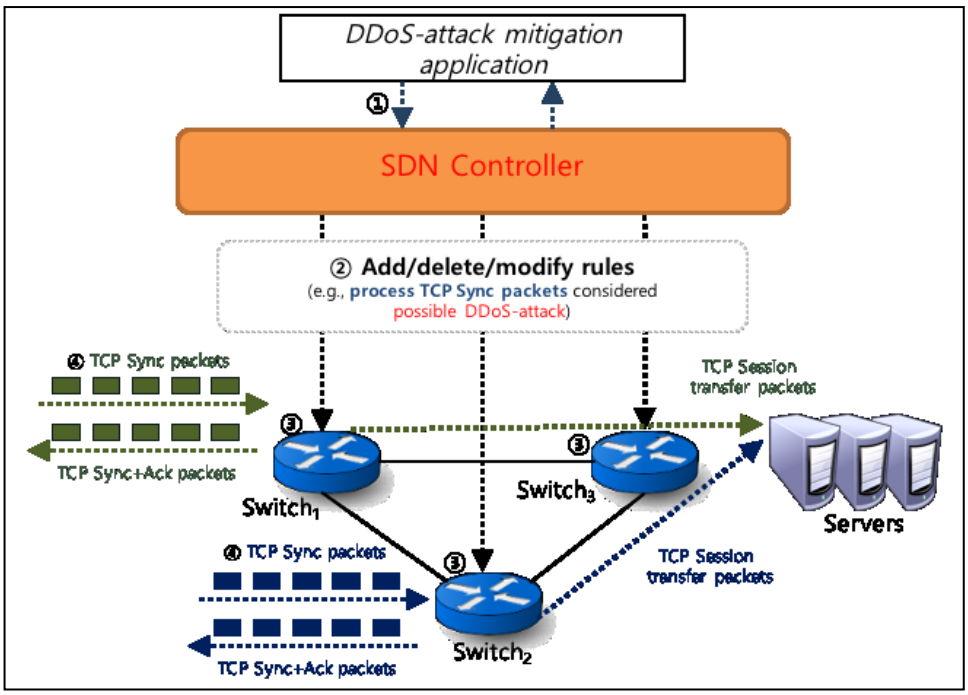


Figure 8-7 – Example scenario for centralized DDoS-attack mitigation for stateful servers

- [Step 1] A mitigator application installs new rules to SDN controller

A DDoS-attack mitigator application should select which switch performs the role of proxy for TCP service. New rule addition is performed by DDoS-attack mitigation application running on top of the SDN controller.

- [Step 2] A SDN controller distributes new rules to appropriate switches

A new rule might be distributed to appropriate switches for DDoSattack mitigation by an SDN controller after installing it. Therefore, the SDN controller sends a flow insert operation that contains the rule (e.g., “Generate TCP Sync+Ack for packets considered DDoS attacks”) to all SDN switches. Therefore, a new rule is installed into the selected switch so that it can generate TCP Sync-Ack packets for TCP Sync as request. If the same requests arrive much more frequently than the expected rate, SDN controller selects new switch so that the switch behaves the role of the server. For the normal TCP Sync, the switch transfers the TCP session to the corresponding server in the private network. It can also be managed centrally such that a SM can determine security policies for their service through a single point, that is, SDN controller.

- [Step 3] All SDN switches apply to new rules in their flow table

All SDN switches add a flow entry dropping future packets considered the DDoS-attack packets to their flow table when receiving the flow insert operation about the DDoS attacks. After that, the SDN switch can generate TCP Sync-Ack packets with a probability proportional to the severity of the DDoS-attack.

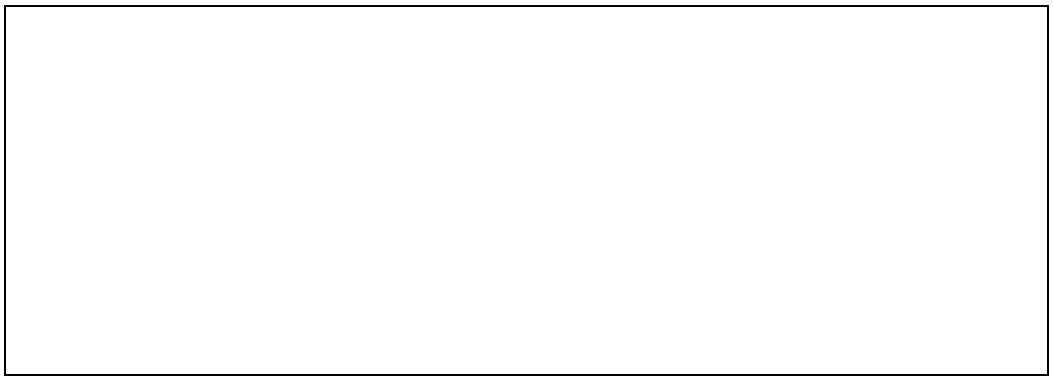
- [Step 4] A SDN switch executes new rules to mitigate DDoS-attack

A SDN switch completely responds to TCP Sync packets from an adversary host randomly when receiving DDoS-attack packets. DDoS-attack requests for stateful are handled by switches instead of actual servers.

8.4 Centralized illegal device management service

8.4.1 Basic concept of centralized illegal device management service

This clause describes the basic concept of centralized illegal device management service. The centralized illegal device manager can manage the blacklist of illegal devices. As shown in Figure 8-8, a centralized illegal device management service manages the list of blacklisted devices to prevent the traffic from those devices. The list of illegal devices is stored in a blacklisting database and can be updated either manually or automatically by independent applications. The centralized illegal device manager periodically loads the list of illegal devices from the blacklisting database and reports those events to the illegal device application which generates new security rules to prevent the network traffic from/to those illegal devices.



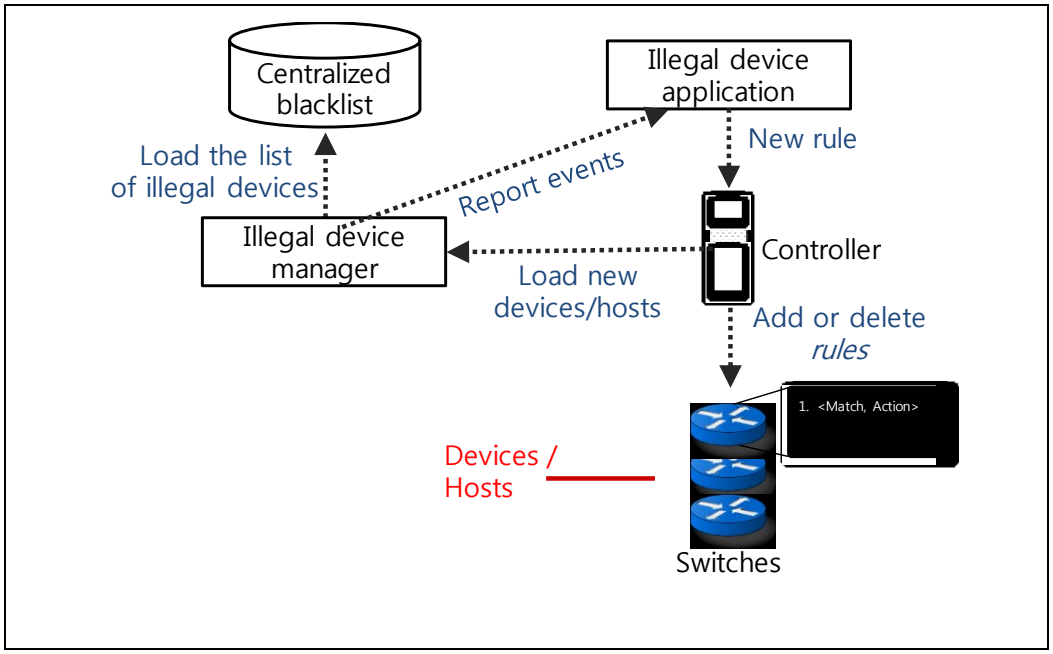


Figure 8-8 – Centralized illegal device management service

8.4.2 Service scenario of centralized illegal device management service

Figure 8-9 shows a centralized illegal device management service for switches. This clause defines processes of preventing traffic from a stolen mobile device. Stopping anyone using the stolen device shows that how SDN controller can manage a centralized illegal device management service. This scenario concentrates on SDN switches.

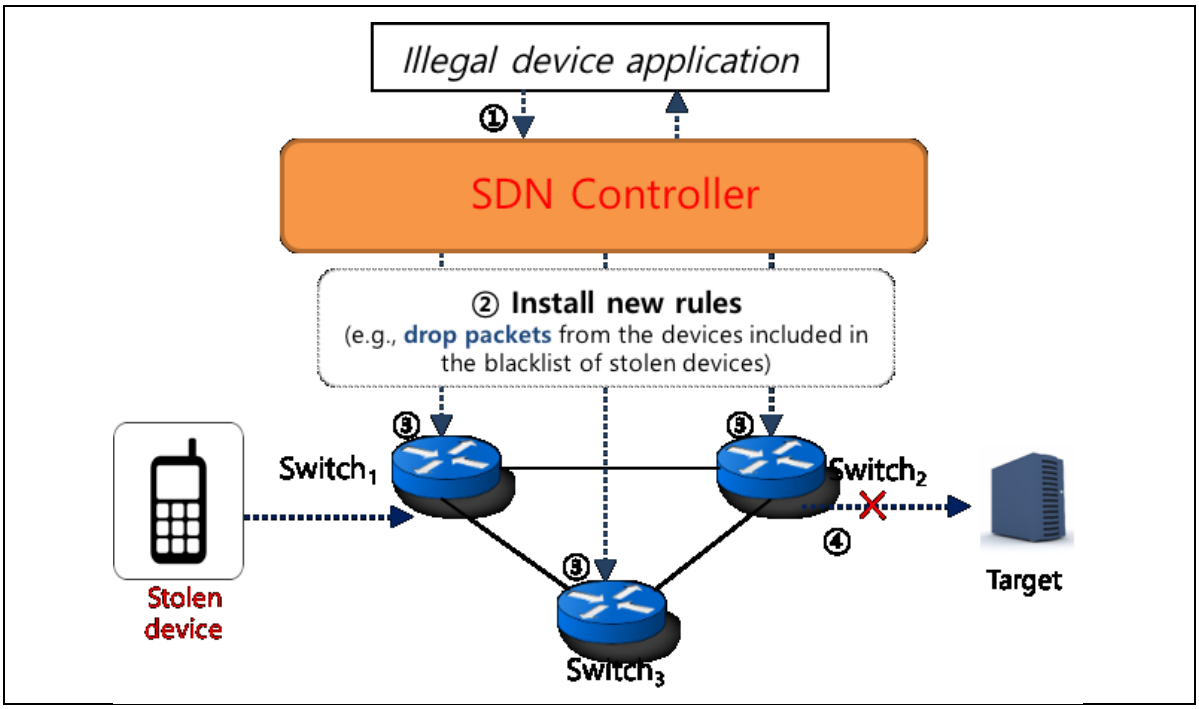


Figure 8-9 –Example scenario for centralized illegal device management service

- [Step 1] Illegal device management application installs new rules

An illegal device application should specify a new rule when the information about new stolen devices is reported from the centralized illegal device manager. As a precondition of this scenario, the illegal device application or SM adds the new rule (e.g., "Drop packets from those devices stored in a centralized blacklist of stolen devices") to the SDN controller.

- [Step 2] A SDN controller distributes new rules

A new rule might be distributed to each switch by an SDN controller after installing it. Therefore, the SDN controller sends a flow insert operation that contains the rule (e.g., "Drop packets from new stolen devices") to all SDN switches. It can also be managed centrally such that a centralized illegal device manager or SM might determine security policies for their service through a single point, that is, SDN controller.

- [Step 3] All SDN switches apply to new rules

All SDN switches add a flow entry dropping future packets from those devices to their flow tables when receiving the flow insert operation about the stolen devices.

- [Step 4] A SDN switch executes new rules

An SDN switch completely drops packets when receiving packets from those devices. Any packets from those devices cannot be passed switch under applied rules.

Note: It is important that the illegal devices are identified. It is required that a unique identity will be used to identify an illegal device by the centralized illegal device manager. At present the SDN controller only identifies the network address such as of a device IP address and MAC address that can be dynamically changed, so a new rule should be installed and the old rule should be deleted on the SDN controller each time the network address of an illegal device is changed.

8.5 Distributed access control management service

8.5.1 Basic concept of distributed access control management service

This clause describes the basic concept of distributed access control management (ACM) service. The ACM module with SDN controller can manage the access right policies hierarchically. As shown in Figure 8-10, an ACM module manages the access rights in order to control the illegal accesses to the resources.

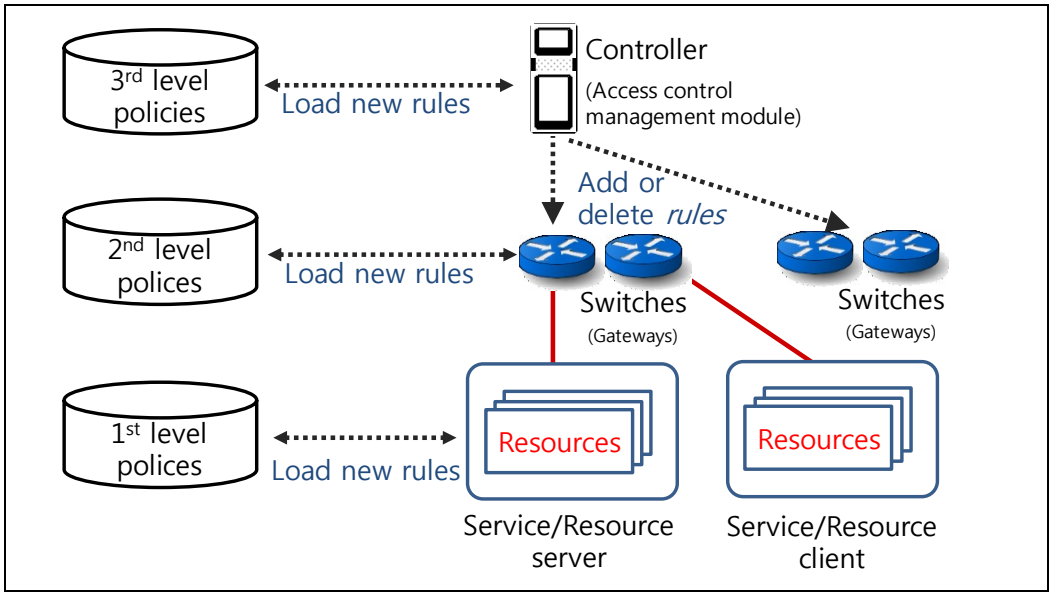


Figure 8-10 – Distributed access control management service

8.5.2 Service scenario of distributed access control management service

Figure 8-11 shows a distributed ACM service for switches. This clause defines processes of getting Service/Resources according to access right policies. This clause shows that how security controller can manage a distributed ACM module service. This scenario concentrates on SDN controller and switches.

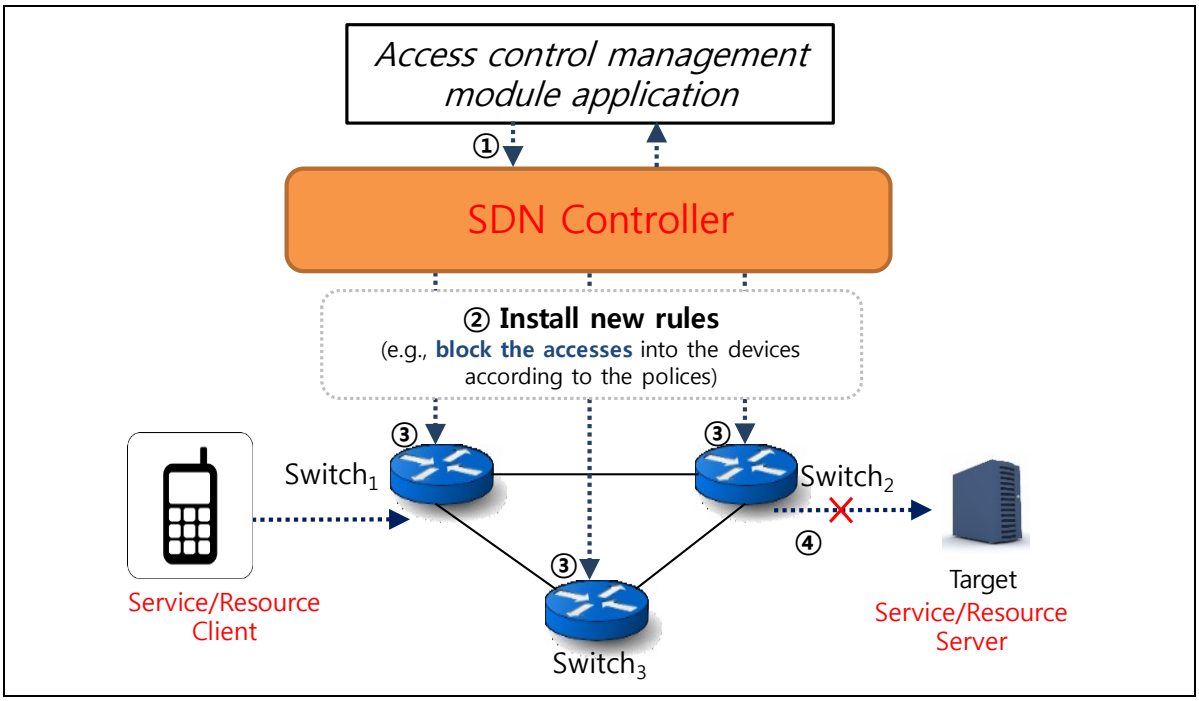


Figure 8-11 –Example scenario for distributed access control management service

- [Step 1] Distributed access control management module installs new policies

A distributed access control management application should specify new policies to get the resources in Service/Resource device (e.g., IoT devices). As a precondition of this scenario, the SM adds the new policies to this ACM application.

- [Step 2] A SDN controller distributes new rules

At first, a new rule or rules should be stored. And then they may be distributed to each switch by an SDN controller. The SDN controller may send an access rights to operate the resource(s) in Service/Resource device. In this case, a SDN controller does not receive any requests from SDN switches for rule distribution. And SDN switches may be able to ask SDN controller to give access rules for resource in Service/Resource devices before.

- [Step 3] All SDN switches apply to new rules

All SDN switches add new rules to their local database to process access authorization requests about Service/Resource devices.

- [Step 4] A SDN switch executes new rules

An SDN switch completely can drop packets when receiving packet from Service/Resource client according to the access rules. Any packets from those clients cannot be passed SDN switch under applied rules. On the other hand, any packets which don’t have any access rules should be informed to the SDN controller in order to manage them by ACM application.

Appendix I

Criteria for Security Services based on SDN

(This appendix does not form an integral part of this Recommendation)

This appendix provides the following criteria for various security services.

I.1 Criteria for security services in intra-domain networks

I.1.1 Centralized firewall service

Legacy firewalls have some challenges such as the expensive cost, performance, management of access control, establishment of policy, and packet-based access mechanism. To address these challenges, this Recommendation presents the framework of a centralized firewall service based on SDN. Firewall rules can be managed flexibly by a centralized server. Existing SDN protocols can be used through standard interfaces between firewall applications and switches.

- Cost

The cost of adding firewalls to network resources such as routers, gateways, and switches is substantial due to the reason that we need to add firewall on each network resource. To solve this, each network resource can be managed centrally such that a single firewall is manipulated by a centralized server.

- Performance

The performance of firewalls is often slower than the link speed of their network interfaces. Every network resource needs to check firewall rules without reference to network conditions. Firewalls can adaptively be deployed depending on network conditions in this framework.

- Management of access control

Sine there may be hundreds of network resources in an administered network, the dynamic management of access control for security services like firewall is a challenge. This is because firewall rules need to be dynamically added for new network attacks.

- Establishment of policy

Policy should be established for each network resource. However, it is difficult to describe what are permitted and denied flows within a specific organization network under management. Thus, a centralized view might be helpful to determine security policies for such a network.

- Packet-based access mechanism

Packet-based access mechanism is not enough in practice since the basic unit of access control is usually users or applications. Therefore, application level rules need to be defined and added to the firewall service by an administrator.

I.1.2 Centralized honeypot service

Legacy honeypots have some challenges such as the expensive cost, performance, management of access control, establishment of policy, and packet-based access mechanism. To address these challenges, this Recommendation presents the framework of a centralized honeypot service based on

SDN. Honeypot places can be managed flexibly by a centralized server. Existing SDN protocols can be used through standard interfaces between honeypot applications and switches.

- Cost

The cost of running additional honeypots in a network is substantial due to the reason that we need to use additional network resources such as hosts for honeypots. To solve this, honeypot places can be managed flexibly by a centralized server.

- Performance

The performance of honeypots depends on the capability of host machines. Every honeypot is always running in the same manner without reference to network and/or attack conditions. Honeypots can adaptively be deployed depending on network and/or attack conditions in this framework.

- Management of access control

Sine there may be hundreds of network resources in an administered network, the dynamic configuration of honeypots is a challenge. This is because honeypot places need to be dynamically changed against new attacks.

- Establishment of policy

Policy should be established for each network resource. However, it is difficult to determine specific honeypot places against suspicious attacks depending on the network and attack conditions. Thus, a centralized view might be helpful to dynamically just security policies over time.

- Honeypot deployment mechanism

Honeypot places should be properly deployed depending on network and attack conditions. It determines the optimal place to monitor and respond to attacks in real time. The honeypot is centrally configured as the intended attack target by a centralized server.

I.2 Criteria for security services in inter-domain networks

I.2.1 Centralized DDoS-attack mitigation service

Centralized DDoS-attack mitigation service defends servers against DDoS attacks outside private network, that is, from public network. The servers are categorized into stateless servers (e.g., DNS servers) and stateful servers (e.g., web servers). Figure 8-3 shows the configuration of DDoS-attack mitigation service in a private network. Switches in the private network are configured in multi-levels, that is, Level-1 switches, Level-2 switches, …, Level-n switches for the dynamic defense lines against a variety of DDoS attacks.

The centralized DDoS-attack mitigation service has some challenges such as the expensive cost, performance, management of access control, establishment of policy, and packet-based access mechanism. To address these challenges, this Recommendation presents the framework of a centralized DDoS-attack mitigation service based on SDN. DDoS-attack mitigation rules can be managed flexibly by a centralized server. Existing SDN protocols can be used through standard interfaces between DDoS-attack mitigator applications and switches.

- Cost

Each network resource can be managed centrally and flexibly with a minimum cost such that switches are configured and manipulated in multi-levels by a centralized server. As the severity

of DDoS attack for a server increases, further-level switches perform selective drops of packets to reduce the impacts of DDoS attacks. That is, suspicious packets for DDoS attack will be dropped earlier at the beginning of the routing path to the victim host.

- Performance

The performance of centralized DDoS-attack mitigation is often slower than the link speed of their network interfaces. In the legacy service, every network resource needs to check DDoS-attack mitigation rules without reference to network conditions. However, DDoS-attack mitigators can adaptively be deployed depending on network conditions in this framework.

- Management of access control

Since there may be hundreds of network resources in an administered network, the dynamic management of access control for security services like DDoS-attack mitigation is a challenge. This is because DDoS-attack mitigation rules need to be dynamically added for new DDoS attacks.

- Establishment of policy

Policy should be established for each network resource. However, it is difficult to determine specific packet drop policies against DDoS attacks depending on the network conditions. Thus, a centralized view might be helpful to dynamically just security policies over time.

- DDoS-attack detection mechanism

DDoS-attack detection is performed by checking whether requests for services from a client come in an expected interval or not. It determines the probability that the requests from a client are DDoS attacks and performs more frequent selective drops of the requests proportionally to the probability.

I.2.2 Centralized illegal device management service

Legacy illegal device management services have some challenges such as the expensive cost, performance, management of access control, establishment of policy, and packet-based access mechanism. To address these challenges, this Recommendation presents a centralized illegal device management service based on SDN. The rules for blacklisting devices can be managed globally. Existing SDN protocols can be used through standard interfaces between firewall applications and switches.

- Cost

The cost of updating blacklists for network resources such as routers, gateways, and switches is substantial due to the reason that we need to update blacklists for each network resource, individually. To solve this, the security rules related to blacklists for each network resource can be managed centrally such that a single illegal device management service is manipulated by a centralized server.

- Performance

Since the packets from the blacklisted devices are dropped at the beginning of the routing path unlike the legacy management service, the performance of the centralized illegal device management service can be improved in practice.

- Management of access control

When blacklists are locally managed, it is not easy to synchronize the locally distributed blacklists since there may be hundreds of network resources in various countries. Security rules need to be dynamically added for new illegal devices.

- Establishment of policy

Policy should be established for each network resource. However, it is difficult to describe what devices are disallowed within a specific organization network under management. Thus, a centralized view might be helpful to determine security policies for such a network.

- Blacklist update mechanism

It is really important to maintain an up-to-date blacklist of illegal devices. Therefore, existing legacy services regularly updating the blacklist database so as to retain the latest information on any illegal devices. In the centralized illegal device management service, the blacklist is centrally managed as a single logical database by a centralized server.

Appendix II

An Example of Packet Data Scan Detection

(This appendix does not form an integral part of this Recommendation)

Packet data scan detection has to be supported in order to detect and mitigate some attacks like worm. The administrator configures the policies to randomly detect some packets of the flow instead of all the packets of the flow for higher performance. There is one possible schema of packet data scan detection [ICIN2015 SDNSEC]: to select the first m consecutive packets from each flow for packet data scan detection. This schema can be designed for all flows or for only flows meeting some conditions, such as packets from a certain source IP address and/or to a certain destination.

OpenFlow™ protocol [b-ONF OpenFlow], as one of SDN southbound interface implementation, may be extended in order to support packet data scan detection. Firstly, two more additional features may be added into the format of the flow entry. These updates have to be reflected into both the controller and switches. One of them is the schema which describes the schema of packet data scan detection. The other is the condition which describes the flows meets some conditions configured by the administrator or applications. Secondly, an optional action (OFPAT_DETECTION) should be added in section of "5.12 Actions" of [b-ONF OpenFlow] as following texts in *italic*: *Optional Action: the Detection action forwards a packet to a specified OpenFlowTM port then to security appliances (e.g., FW, IDP, DPI, etc) for further data scan detection.* This new action is similar to the action OFPAT_OUTPUT in OpenFlow™ protocol. Finally, action structures should be updated in section "7.2.4 Action Structures" of [b-ONF OpenFlow] as below with texts in *italic*:

```
enum ofp_action_type {
OFPAT_OUTPUT = 0,                                   /* Output to switch port. */
OFPAT_DETECTION = XX (a given number),      /*Output to switch port */
OFPAT_COPY_TTL_OUT = 11,      /* Copy TTL "outwards" – from
                                 next-to-outermost to outermost */
OFPAT_COPY_TTL_IN = 12,       /* Copy TTL "inwards" – from
                                outermost to next-to-outermost */
OFPAT_SET_MPLS_TTL = 15,      /* MPLS TTL */
OFPAT_DEC_MPLS_TTL = 16,      /* Decrement MPLS TTL */
OFPAT_PUSH_VLAN = 17,         /* Push a new VLAN tag */
OFPAT_POP_VLAN = 18,          /* Pop the outer VLAN tag */
OFPAT_PUSH_MPLS = 19,         /* Push a new MPLS tag */
OFPAT_POP_MPLS = 20,          /* Pop the outer MPLS tag */
OFPAT_SET_QUEUE = 21,         /* Set queue id when outputting to a port */
OFPAT_GROUP = 22,             /* Apply group. */
OFPAT_SET_NW_TTL = 23,        /* IP TTL. */
OFPAT_DEC_NW_TTL = 24,        /* Decrement IP TTL. */
OFPAT_SET_FIELD = 25,         /*Set a header field using OXM TLV format*/
OFPAT_PUSH_PBB = 26,          /* Push a new PBB service tag (I-TAG) */
OFPAT_POP_PBB = 27,           /* Pop the outer PBB service tag (I-TAG) */
OFPAT_EXPERIMENTER = 0xffff
};
A Detection action uses the following structure and fields:
/*Action structure for OFPAT_DETECTION which sends packets out 'port'.*/
struct ofp_action_detection {
uint16_t type;                /* OFPAT_DETECTION. */
uint16_t len;                 /* Length is 16. */
```

```
    uint32_t port;                  /* Output port. */
    uint16_t schema;                /* One possible schema is: to select the first m
                                       consecutive packets from each flow. */
    uint32_t condition;             /* One possible condition: packets
                                       of the flow to a certain destination . */
    };
OFP_ASSERT(sizeof(struct ofp_action_output) == 10);
```

Bibliography

[b-ITU-T X.1252] Recommendation ITU-T X.1252 (2010), Baseline identity management terms and definitions

[b-ITU-T X.800] Recommendation ITU-T X.800 (1991), Security architecture for open systems interconnection for CCITT applications

[b-ICIN2015 SDNSEC] Z. Hu, M. Wang, X. Yan, Y. Yin and Z. Luo, “A Comprehensive Security Architecture for SDN”, 18th International Conference on Intelligence in Next Generation Networks, IEEE, 2015, pp30-37

[b-ONF OpenFlow] "OpenFlow Switch Specification Version 1.4.0", Open Networking Foundation. Available: https://www.opennetworking.org/sdn-resources/onf-specifications/openflow

_____