Network Working Group
Internet Draft
Intended status: Informational
Expires: November 2015

L. Dunbar
Huawei
M. Zarny
Goldman Sachs
C. Jacquenet
M. Boucadair
France Telecom
S. Chakrabarty
US Ignite

May 28, 2015

Interface to Network Security Functions (I2NSF) Problem Statement
draft-dunbar-i2nsf-problem-statement-05.txt

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79. This document may not be modified, and derivative works of it may not be created, except to publish it as an RFC and to translate it into languages other than English.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

This Internet-Draft will expire on November 28, 2015.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Abstract

This document describes the motivation and the problem statement for Interface to Network Security Functions (I2NSF).

Table of Contents

1. Introduction.....3
2. Requirements Language.....4
3. Problem Space.....5
3.1. Challenges Facing Security Service Providers.....5
3.1.1. Diverse types of Security Functions.....5
3.1.2. Diverse Interfaces to Control NSFs.....6
3.1.3. Diverse Interface to monitor the behavior of NSFs....7
3.1.4. More Distributed NSFs and vNSFs.....7
3.1.5. More Demand to Control NSFs Dynamically.....7
3.1.6. Demand for multi-tenancy to control and monitor NSFs.7
3.1.7. Lack of Characterization of NSFs and Capability Exchange.....7
3.1.8. Lack of mechanism for NSFs to utilize external profiles 8
3.2. Challenges Facing Customers.....9
3.2.1. NSFs from heterogeneous administrative domains.....9
3.2.2. Today's Control Requests are Vendors Specific.....9

3.2.3. Difficulty to Monitor the Execution of Desired Policies 11
3.3. Difficulty to Validate Policies across Multiple Domains..11
3.4. Lack of Standard Interface to Inject Feedback to NSF.....12
3.5. Lack of Standard Interface for Capability Negotiation....12
4. Scope of the proposed work.....12
5. Other Potential Uses of I2NSF.....14
6. Related Industry Initiatives.....14
6.1. Related IETF WGs.....14
6.2. Relationship with ETSI NFV ISG.....16
6.3. OpenStack Firewall/Security as a Service.....16
6.4. Security as a Service by Cloud Security Alliance.....17
7. Manageability Considerations.....17
8. Security Considerations.....17
9. IANA Considerations.....17
10. References.....17
10.1. Normative References.....17
10.2. Informative References.....17
11. Acknowledgments.....19
11.1. Appendix: Relationship with Open Source Communities.....20

1. Introduction

This document describes the motivation and the problem space for the Interface to Network Security Functions (I2NSF) effort.

The growing challenges and complexity in maintaining a secure infrastructure, complying with regulatory requirements, and controlling costs are enticing enterprises into consuming network security functions hosted by service providers. The hosted security service is especially attractive to small and medium size enterprises who suffer from a lack of security experts to continuously monitor, acquire new skills and propose immediate mitigations to ever increasing sets of security attacks.

According to [Gartner-2013], the demand for hosted (or cloud-based) security services is growing. Small and medium-sized businesses (SMBs) are increasingly adopting cloud-based security services to replace on-premises security tools, while larger enterprises are deploying a mix of traditional and cloud-based security services.

To meet the demand, more and more service providers are providing hosted security solutions to deliver cost-effective managed security services to enterprise customers. The hosted security services are primarily targeted at enterprises (especially small/medium ones), but could also be provided to any kind of mass-market customer.

As the result, the Network security functions (NSFs) are provided and consumed in increasingly diverse environments. Users of NSFs could consume network security services hosted by one or more providers, which may be their own enterprise, service providers, or a combination of both.

This document does not elaborate on specific use case. The reader should refer to [I2NSF-ACCESS], [I2NSF-DC] and [I2NSF-Mobile] for a more in-depth discussion on the I2NSF use cases.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119 [RFC2119].

This document makes use of the following terms and acronyms:

DC: Data Center

Network Security Function (NSF): functions to ensure integrity, confidentiality and availability of network communications, to detect unwanted activity, and to block it or at least mitigate its effects on the network.

Hosted security function: Refers to a security function that it is hosted by another network.

Flow-based Network Security Function: A function that inspects network flows according to a policy intended for

enforcing security properties. Flow-based security also means that packets are inspected in the order they are received, and without modification to the packet due to the inspection process (MAC rewrites, TTL decrement action; even NAT would be outside the inspection process).

3. Problem Space

The following sub-sections describe the problems and challenges facing customers and security service providers (called service provider, for short) when security functions are no longer physically hosted by customer's administrative domain.

The "Customer-Provider" relationship may be between any two parties: different firms or different domains of the same firm. Contractual agreements may be required in such contexts to formally document the customer's security requirements and the provider's guarantees to fulfill those requirements. Such agreements may detail protection levels, escalation procedure, alarms reporting, etc. There is currently no standard mechanism to capture those requirements.

Note a service provider may be a customer of another service provider.

3.1. Challenges Facing Security Service Providers

3.1.1. Diverse types of Security Functions

There are many types of NSFs. NSFs by different vendors can have different features and have different interfaces. NSFs can be deployed in multiple locations in a given network, and perhaps have different roles.

Below are a few examples of security functions and locations or contexts in which they are often deployed:

External Intrusion & Attack Protection:
e.g., Firewall/ACL; Authentication; IPS; IDS; Endpoint Protection; etc;

Security Functions in a DMZ:
e.g., Firewall/ACL; IDS/IPS, authentication and authorization services, NAT, forward proxies, application FWs, AAA; etc.

Internal Security Analysis & report:
e.g., Security Log; Event Correlation; Forensic Analysis; etc;

Internal Data and Content Protection:
e.g., Encryption; Authorization; Public/Private key management for internal database, etc.

Given the diversity of security functions, contexts in which they can be deployed, and constant evolution of these functions, standardizing all aspects of security functions is challenging, most probably not feasible, and not necessary. For example, from an I2NSF perspective, there is no need to standardize on how a firewall filters are created or applied. What is needed is having an interface to control and monitor the behavior of NSFs.

3.1.2. Diverse Interfaces to Control NSFs

To provide effective and competitive solutions and services, Security Service Providers may need to utilize multiple security functions from various vendors to enforce the security policies desired by their customers.

Yet because no widely accepted industry standard security interfaces exist today, management of NSFs (device and policy provisioning, monitoring, etc.) tends to be bespoke, essentially as offered by product vendors. As a result, automation of such services, if it exists at all, is also bespoke. It is worth noting that even with the traditional way of deploying security features, there is still a gap to coordinate among implementations from distinct vendors. This is mainly the reason why mono-vendor security functions are enabled in a given network segment.

3.1.3. Diverse Interface to monitor the behavior of NSFs

Obviously, enabling a security function (e.g., firewall [I-D.ietf-opsawg-firewalls]) does not mean that a network is protected. As such, it is necessary to have a mechanism to monitor the execution status of NSFs.

3.1.4. More Distributed NSFs and vNSFs

The security functions that are invoked to enforce a security policy can be located in different equipment and network locations.

The European Telecommunications Standards Institute (ETSI) Network Function Virtualization (NFV) initiative creates new management challenges for security policies to be enforced by distributed, virtual, network security functions (vNSF).

vNSF has higher risk of failure, migrating, and state changes as their hosting VMs being created, moved, or decommissioned.

3.1.5. More Demand to Control NSFs Dynamically

In the advent of SDN [SDN-Security], more clients, applications or application controllers need to dynamically update their communication policies that are enforced by NSFs. The Security Service Providers have to dynamically update control requests to NSFs upon receiving the requests from their clients.

3.1.6. Demand for multi-tenancy to control and monitor NSFs.

Service providers may require having several operational units to control and monitor the NSFs, especially when NSFs become distributed and virtualized.

3.1.7. Lack of Characterization of NSFs and Capability Exchange

To offer effective security services, service providers need to activate various security functions manufactured by multiple

vendors. Even within one product category (e.g., firewall), security functions provided by different vendors can have different features and capabilities: filters that can be designed and activated by a firewall may or may not support IPv6, depending on the firewall technology, for example.

Service Provider management system (or controller) needs ways to retrieve the capabilities of service functions by different vendors so that it could build an effective security solution.

These capabilities can be documented in a static manner or via an interface for security functions vendors to register to service provider security management system. This dynamic capability registration is useful for automation because security functions may be subject to software and hardware updates. These updates may have implications on the policies enforced by the NSFs.

Today, there is no standard method for vendors to describe the capabilities of their security functions. Without a common technical framework to describe the capabilities of security functions, service providers can’t automate the process of selecting NSFs by different vendors to accommodate customer’s requirements.

3.1.8. Lack of mechanism for NSFs to utilize external profiles

Many security functions depend on signature files or profiles to perform, e.g. IPS/IDS Signatures. Different policies might need different signatures or profiles. Today, most vendors have their vendor specific signatures or profiles. As the industry moves towards more open environment, sharing profile or black database can be win-win strategy for all parties involved. There might be Open Source provided signature/profiles (e.g. by Snort or others) in the future.

There is a need to have a standard envelop (i.e. the format) to allow NSFs to use external profiles.

3.2. Challenges Facing Customers

When customers invoke hosted security services, their security policies may be enforced by a collection of security functions hosted in different domains. Customers may not have security skills. As such, they may not be able to express sufficiently precise requirements or security policies. Usually these customers express expectations (that can be viewed as loose security requirements). Customers may also express guidelines such as which critical communications are to be preserved during critical events, which hosts are to service even during severe security attacks, etc.

3.2.1. NSFs from heterogeneous administrative domains

Many medium and large enterprises have deployed various on-premises security functions which they want to continue to use. They are looking for combining local security functions with remote hosted security functions to achieve more efficient and immediate counter-measures to both Internet-originated attacks and enterprise network-originated attacks.

Some enterprises may only need the hosted security services for their remote branch offices where minimal security infrastructures/capabilities exist. The security solution can consist of NSFs on customer networks and NSFs on service provider networks.

3.2.2. Today’s Control Requests are Vendors Specific

Customers may consume NSFs by multiple service providers. Customers need to express their security requirements, guidelines, and expectations to the service providers, which in turn will be translated into security policies and associated configuration sets to the set of security functions. But no standard technical characterization and/or APIs exist, even for most common security

services. Most security services are accessible only through disparate, proprietary interfaces (e.g., portals, APIs), in whatever format vendors choose to offer.

Without standard interfaces it is complex for customers to update security policies and integrate with services provided by the security service providers. This complexity is induced by the diversity of the configuration models, policy models, supported management interfaces, etc.

The current practices that rely on the use of scripts that generates automatically scripts have to be adjusted each time an implementation from a different vendor is enabled in a provider side.

Customers may also require means to easily update/modify their security requirements with immediate effect in the underlying involved NSFs.

While security agreements are in place, security functions may be solicited without requiring an explicit invocation means. Nevertheless, some explicit invocation means may be required to interact with a service function.

Here is an example of how standard interfaces could help achieve faster implementation time cycles. Let us consider a customer who would like to dynamically allow an encrypted flow with specific port, src/dst addresses or protocol type through the firewall/IPS to enable an encrypted video conferencing call only during the time of the call. With no commonly accepted interface in place, the customer would have to learn about the particular provider’s firewall/IPS interface, and send the request in the provider’s required format. If a firewall/IPS interface standard exists, the customer would be able to send the request, without having to do much preliminary legwork. Such a standard helps providers too since they could now offer the same firewall/IPS interface to represent firewall/IPS services, which may be offered by different vendors’ products. They have now abstracted the firewall/IPS services. Lastly, it helps the firewall/IPS vendors since they could now work on common specifications.

3.2.3. Difficulty to Monitor the Execution of Desired Policies

How a policy is translated into technology-specific actions is hidden from the customers. However, customers still need ways to monitor the delivered security service that is the result of the execution of their desired security requirements, guidelines and expectations.

Today, there is no standard way for customers to get security service assurance (including running "what-if" scenarios to assess the efficiency of the delivered security service) of their specified security policies properly enforced by the security functions in the provider domain.

3.3. Difficulty to Validate Policies across Multiple Domains

One key aspect of a hosted security service with security functions located at different premises is to have a standard interface to express, monitor and verify security policies that combine several distributed security functions. This becomes more crucial when NSFs are instantiated in Virtual Machines because NSFs can be more distributed and sometimes multiple NSFs are combined together to perform one task.

Without standard interfaces and security policy data models, the enforcement of a customer-driven security policy remains challenging because of the inherent complexity brought by the combined invocation of several, yet vendor-specific security functions, but also because of the accompanying complexity of configuration procedures and operational tasks in a multi-vendor, heterogeneous environment.

Ensuring the consistent enforcement of the policies at various domains is challenging. Standard data models are likely to contribute to softening that issue.

3.4. Lack of Standard Interface to Inject Feedback to NSF

Today, many security functions, such as IPS and Antivirus, depend heavily on the associated profiles. They can perform more effective protection if they have the up-to-date profiles. As more sophisticated threats arise, enterprises, vendors, and service providers have to rely on each other to achieve optimal protection. [CA] is one of those initiatives that aim at combining efforts conducted by multiple organizations.

Today there is no standard interface to exchange security profiles between organizations.

3.5. Lack of Standard Interface for Capability Negotiation

There could be situations when the NSFs selected can’t perform the policies from the Security Controller, due to resource constraints. To support the automatic control in the SDN-era, it is necessary to have a set of messages for proper negotiation between the Security Controller and the NSFs.

4. Scope of the proposed work

The primary goal of I2NSF is to define an information model, a set of software interfaces and data models for controlling and monitoring aspects of physical and virtual NSFs. Other aspects of NSFs, such as device or network provisioning and configuration, are out of scope. Controlling and monitoring of NSFs should include the ability to specify, query, monitor, and control the NSFs by one or more management entities. Since different security vendors support different features and functions on their devices, I2NSF will focus on flow-based NSFs that provide treatment to packets/flows, such as IPS/IDS, Web filtering, flow filtering, deep packet inspection, or pattern matching and remediation.

There are two layers of interfaces envisioned in the I2NSF approach:

- The I2NSF Capability Layer specifies how to control and monitor NSFs at a functional implementation level. That is, I2NSF will standardize a set of interfaces by which control and management of NSFs may be invoked, operated, and monitored. (I2NSF will not work on any other aspects of NSFs. Nor will I2NSF at this stage specify how to derive control and monitoring capabilities from higher level security policies for the Capability Layer.)
- The I2NSF Service Layer defines how clients' security policies may be expressed and monitored. The Service Layer is out of scope for this phase of I2NSF's work. However, I2NSF will provide a forum for Informational drafts on data models, APIs, etc. that demonstrate how service layer policies may be translated to Capability Layer functions.

The concrete work at the I2NSF Capability Layer includes development of

- An information model that defines concepts required for standardizing the control and monitoring of NSFs.
- A set of YANG data models, derived from the above information model.
- The capability registry (IANA) that enables the characteristics and behavior of NSFs to be specified using a vendor-neutral vocabulary without requiring the NSFs themselves to be standardized. The registry enables various mechanisms, including policy rules, to be used to match monitor and control functions to the needs of an application and/or environment.
- The proper secure communication channels to carry the controlling and monitoring information between the NSFs and their management entity (or entities).

Standard interfaces for monitoring and controlling the behavior of NSFs are essential building blocks for Security Service Providers to automate the use of different NSFs from multiple vendors by their Security management entities. This work will leverage the existing protocols and data models defined by I2RS, Netconf, and NETMOD.

I2NSF may be invoked by any (authorized) client-e.g., upstream applications (controllers), orchestration systems, security portals, etc.

5. Other Potential Uses of I2NSF

The I2NSF framework allows the clients to view, request, and/or verify the security functions/policies offered by providers at different premises. This framework can make it possible for a cluster of devices requiring the similar security policies to have consistent policies across multiple sites.

Network service providers can provide "Hosted Security Functions" services. Network providers can also act as security function brokers to facilitate if not optimize the enforcement of customer-driven security policies. They can expose a service catalog and standard mechanisms by which enterprises (or applications) can query, request, or/and verify the needed security functions or policies.

With the standard interfaces for clients to request the required security functions and policies, network operators can leverage their current service to enterprises (e.g. VPN, private IP services) and access to a vast population of end users to offer a set of consolidated Security solutions and policies. Network operators can be instrumental in defining a common interface and framework as part of an IETF-conducted specification effort.

6. Related Industry Initiatives

6.1. Related IETF WGs

IETF NETCONF: I2NSF should consider using the NETCONF protocol exchange security policy provisioning information between participating devices/security functions and the computation logic (a.k.a., a security Policy Decision Point (PDP)) that resides in the control plane and which makes the decisions to dynamically allocate resources and enforce customer-driven security policies.

NETMOD ACL Model: [I-D.ietf-netmod-acl-model] describes the very basic attributes for access control. I2NSF will extend the ACL data model to be more comprehensive, for example, extend to multiple actions and policies, and describes various services associated with the security functions under consideration.

In addition, I2NSF has to specify ways to monitor/report of Packet Based Security Functions.

I2RS: the WG currently discusses the specification of an interface between the forwarding and the control planes, to facilitate the dynamic enforcement of traffic forwarding policies based upon IGP/BGP route computation results. I2NSF is looking specifically into expressing security policies in two layers. I2NSF should leverage the protocols and data models developed by I2RS.

I2NSF aims to develop the additional information models and data models for distributed security functions, like the firewall and IPS/IDS. The policy structure specified by [I-D.hares-i2rs-bnp-info-model] can be used by I2NSF to be extended to include recursive actions to other security functions.

The IETF SFC WG specifies service function chaining techniques while treating service functions as a black box; VNFpool is about the reliability and availability of the virtualized network functions. But neither addresses how service functions are invoked, or configured.

Both SFC and VNFpool do not cover in-depth specification (e.g. rules for the requested FW) to invoke security functions. In SFC and VNFpool, a firewall function is a black box that is treated in the same way as a video optimization function. SFC and VNFpool do not cover the negotiation part, e.g. Client needs Rules x/y/z for FW, but the Provider can only offer x/z.

The IETF SACM (Security Assessment and Continuous Monitoring) WG specifies mechanisms to assess endpoint security. The endpoints can be routers, switches, clustered DB, or an installed piece of software. SACM is about "How to encode that policy in a manner where assessment can be automated". For example:

- a Solaris 10 SPARC or Windows 7 system used in an environment that requires adherence to a policy of Mission Critical Classified,
- rules like "The maximum password age must be 30 days" and "The minimum password age must be 1 day"

[I2NSF-GAP] has a more extensive study comparing I2NSF with various existing efforts in similar/adjacent areas.

6.2. Relationship with ETSI NFV ISG

ETSI's NFV ISG defines the architecture to pool together many virtual network functions to be managed and consumed collectively.

I2NSF is one of the enabling tools for NFV, specifically the VNF as a Service (VNFaaS) specified by ETSI NFV Group Specification Use Cases [gs_NFV].

ETSI's NFV ISG effort is actively contributed by service providers. It defines a detailed service model for VNFaaS as well as requirements that should be taken into account by the I2NSF initiative.

6.3. OpenStack Firewall/Security as a Service

Open source projects like OpenStack and CloudStack have begun to tackle the issues of interfaces to security functions but much work remains.

OpenStack completed the Firewall as a Service project and specified the set of APIs for Firewall services [API]

OpenStack has defined the APIs for managing Security Groups [SG]

The attributes defined by OpenStack Firewall/Security as a Service are at this point are basic. However, they can serve as the basis of the information model that the I2NSF IETF initiative aims to specify.

6.4. Security as a Service by Cloud Security Alliance

https://cloudsecurityalliance.org/research/secaas/#_get-involved

SaaS by CSA is at the initial stage of defining the scope of work.

7. Manageability Considerations

Management of NSFs usually include configuration of devices, signaling and policy provisioning. I2NSF will only focus on the policy provisioning part.

8. Security Considerations

Having a secure access to control and monitor NSFs is crucial for hosted security service. Therefore, proper secure communication channels have to be carefully specified for carrying the controlling and monitoring information between the NSFs and their management entity (or entities).

9. IANA Considerations

This document requires no IANA actions. RFC Editor: Please remove this section before publication.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

10.2. Informative References

[SG] http://docs.openstack.org/admin-guide-cloud/content/securitygroup_api_abstractions.html

[API] http://docs.openstack.org/admin-guide-cloud/content/fwaas_api_abstractions.html

[CA] http://cyberthreatalliance.org/

[I-D.hares-i2rs-bnp-info-model] Hares, S., Wu, Q., Tantsura, J., and R. White, "An Information Model for Basic Network Policy and Filter Rules", draft-hares-i2rs-bnp-info-model-02 (work in progress), March 2015.

[I-D.ietf-netmod-acl-model] Bogdanovic, D., Sreenivasa, K., Huang, L., and D. Blair, "Network Access Control List (ACL) YANG Data Model", draft-ietf-netmod-acl-model-02 (work in progress), March 2015.

[I-D.ietf-opsawg-firewalls] Baker, F. and P. Hoffman, "On Firewalls in Internet Security", draft-ietf-opsawg-firewalls-01 (work in progress), October 2012.

[RFC7297] Boucadair, M., "IP Connectivity Provisioning Profile", RFC7297, April 2014.

[I2NSF-PACKET] E. Lopez, "Packet-based Paradigm for Interfaces to NSFs", <draft-lopez-i2nsf-packet-00>, March 2015.

[I2NSF-ACCESS] A. Pastor, et al, "Access Use Cases for an Open OAM Interface to Virtualized Security Services", <draft-pastor-i2nsf-access-usecases-00>, Oct 2014.

[I2NSF-DC] M. Zarny, et al, "I2NSF Data Center Use Cases", <draft-zarny-i2nsf-data-center-use-cases-00>, Oct 2014.

[I2NSF-MOBILE] M. Qi, et al, "Integrated Security with Access Network Use Case", <draft-qi-i2nsf-access-network-usecase-00>, Oct 2014.

[SDN-Security] J. Jeong, et al, "Requirement for Security Services based on Software-Defined Networking", <draft-jeong-i2nsf-sdn-security-services-01>, March 2015.

[I2NSF-GAP] D. Zhang, et al, "Analysis of Existing Work for I2NSF", <draft-zhang-gap-analysis-00>, Feb 2015.

[gs_NFV] ETSI NFV Group Specification, Network Functions Virtualizsation (NFV) Use Cases. ETSI GS NFV 001v1.1.1, 2013.

[Gartner-2013] E. Messmer, "Gartner: Cloud-based security as a service set to take off", Network World, 31 October 2013

[NW-2011] J. Burke, "The Pros and Cons of a Cloud-Based Firewall", Network World, 11 November 2011

[Application-SDN] J. Giacomonni, "Application Layer SDN", Layer 123 ONF Presentation, Singapore, June 2013

11. Acknowledgments

Acknowledgments to Diego Lopez, Ed Lopez, Andy Malis, John Strassner, and many others for review and contribution to the content.

This document was prepared using 2-Word-v2.0.template.dot.

Authors' Addresses
Linda Dunbar
Huawei Technologies
5340 Legacy Drive, Suite 175
Plano, TX 75024, USA
Phone: (469) 277 5840
Email: ldunbar@huawei.com

Myo Zarny
Goldman Sachs
30 Hudson Street
Jersey City, NJ 07302
Email: myo.zarny@gs.com

Christian Jacquenet
France Telecom
Rennes 35000
France
Email: Christian.jacquenet@orange.com

Mohamed Boucadair
France Telecom
Rennes 35000
France
Email: mohamed.boucadair@orange.com

Shaibal Chakrabarty
US Ignite
1776 Massachusetts Ave NW, Suite 601
Washington, DC 20036
Phone: (214) 708 6163
Email: shaibalc@us-ignite.org

11.1. Appendix: Relationship with Open Source Communities

One of the goals of the I2NSF initiative is to form a collaborative loop from IETF to Industry Open Source Communities.

Open-source initiatives are not to be considered as an alternative to formal standardization processes. On the contrary, they are complementary, with the former acting as an enabler and accelerator of the latter. Open-source provides an ideal mechanism to quick prototyping and validating contending proposals, and demonstrating the feasibility of disruptive ideas that could otherwise not be considered. In this respect, open-source facilitates the engagement in the standardization process of small (and typically more dynamic) players such as start-ups and research groups, which would see better opportunities of being heard and a clearer rewards to their efforts. An open-source approach is extremely useful as well for the production of open reference implementations of the standards at the same (or even faster) pace they are defined. The availability of such reference implementations translate into much simpler interoperability and conformance assessments for both providers and users, and can become the basis for incremental differentiation of a common solution, thus allowing a cooperative competition ("coopetition") model.

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 6, 2017

J. Jeong
H. Kim
Sungkyunkwan University
J. Park
ETRI
T. Ahn
S. Lee
Korea Telecom

July 5, 2016

Software-Defined Networking Based Security Services using Interface to Network Security Functions
draft-jeong-i2nsf-sdn-security-services-05

Abstract

This document describes a framework, objectives, requirements, and use cases for security services based on Software-Defined Networking (SDN) using a common Interface to Network Security Functions (I2NSF). It first proposes the framework of SDN-based security services in the I2NSF framework. It then explains three use cases, such as a centralized firewall system, centralized DDoS-attack mitigation system, and centralized VoIP/VoLTE security system.

Status of This Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on January 6, 2017.

Copyright Notice

Copyright (c) 2016 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 3
2. Requirements Language 4
3. Terminology 4
4. Overview 5
5. Objectives 7
6. Requirements 8
7. Use Cases 9
7.1. Centralized Firewall System 9
7.2. Centralized DDoS-attack Mitigation System 10
7.3. Centralized VoIP/VoLTE Security System 12
8. Security Considerations 14
9. Acknowledgements 14
10. References 14
10.1. Normative References 14
10.2. Informative References 15
Appendix A. Changes from draft-jeong-i2nsf-sdn-security-services-04 16

1. Introduction

Software-Defined Networking (SDN) is a set of techniques that enables users to directly program, orchestrate, control and manage network resources through software (e.g., SDN applications). It relocates the control of network resources to a dedicated network element, namely SDN controller. The SDN controller uses interfaces to arbitrate the control of network resources in a logically centralized manner. It also manages and configures the distributed network resources, and provides the abstracted view of the network resources to the SDN applications. The SDN applications can customize and automate the operations (including management) of the abstracted network resources in a programmable manner via the interfaces [RFC7149][ITU-T.Y.3300][ONF-OpenFlow][ONF-SDN-Architecture].

Due to the increase of sophisticated network attacks, the legacy security services become difficult to cope with such network attacks in an autonomous manner. 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 document describes a framework, objectives and requirements to support the protection of network resources through SDN-based security services using a common interface to Network Security Functions (NSF) [i2nsf-framework]. It uses an interface to NSF (I2NSF) for such SDN-based security services that are performed in virtual machines through network functions virtualization [ETSI-NFV].

This document addresses the challenges of the exisiting systems for security services. As feasible solutions to handle these challenges, this document proposes three use cases of the security services, such as a centralized firewall system, centralized DDoS-attack mitigation system, and centralized VoIP/VoLTE security system.

For the centralized firewall system, this document raises limitations in the 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 document introduces a use case of SDN-based firewall system to overcome these limitations.

For the centralized DDoS-attack mitigation system, this document raises limitations in the 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 document introduces a use case of SDN-based DDoS-attack mitigation system to provide an autonomous and prompt configuration for suspicious network traffic.

For the centralized VoIP/VoLTE security system, this documents raises challenges in the legacy VoIP/VoLTE security system in terms of provisioning time, the granularity of security, cost, and the establishment of policy. This document shows a use case of SDN-based VoIP/VoLTE security system to resolve these challenges along in the I2NSF framework.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

3. Terminology

This document uses the terminology described in [RFC7149], [ITU-T.Y.3300], [ONF-OpenFlow], [ONF-SDN-Architecture], [ITU-T.X.1252], and [ITU-T.X.800]. In addition, the following terms are defined below:

- Software-Defined Networking: 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 [ITU-T.Y.3300].

- Access Control: 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 [ITU-T.X.1252].

- Access Control Policy: The set of rules that define the conditions under which access may take place [ITU-T.X.800].

- Access Control Policy Rules: Security policy rules concerning the provision of the access control service [ITU-T.X.800].

- 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.

- 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.

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

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

- Centralized VoIP/VoLTE Security System: A centralized security system that handles the security issues related to VoIP and VoLTE services. SDN can work as a network-based security system through a standard interface between VoIP/VoLTE security applications and network resources.

4. Overview

This section describes the referenced architecture to support SDN-based security services, such as centralized firewall system and centralized DDoS-attack mitigation system. Also, it describes a framework for SDN-based security services using I2NSF.

As shown in Figure 1, network security functions (NSFs) as security services (e.g., firewall, DDoS-attack mitigation, VoIP/VoLTE, web filter, and deep packet inspection) run on the top of SDN controller [ITU-T.Y.3300] [ONF-SDN-Architecture]. When an administrator enforces security policies for such 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.

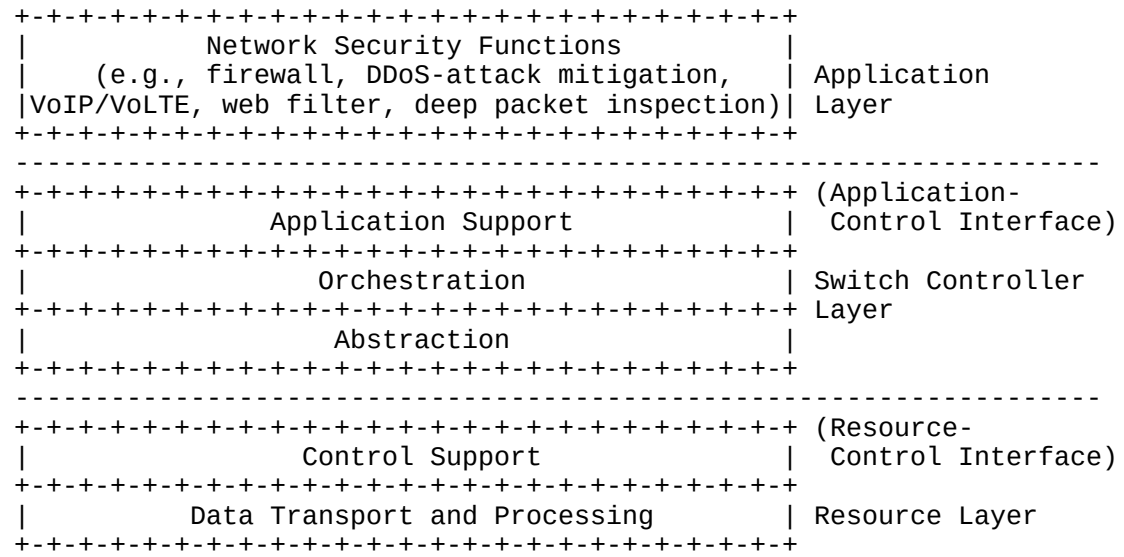


Figure 1: High-level Architecture for SDN-based Security Services

Figure 2 shows a framework to support SDN-based security services using I2NSF [i2nsf-framework]. As shown in Figure 2, I2NSF client can use security services by delivering their high-level security policies to security controller via client facing interface. Security controller asks NSFs to perform function-level security services via NSF facing interface. The NSFs run on top of virtual machines through Network Functions Virtualization (NFV) [ETSI-NFV]. NSFs ask switch controller to perform their required security services on switches under the supervision of switch controller. In addition, security controller uses registration interface to communicate with developer’s management system for registering (or deregistering) the developer’s NSFs into (or from) the NFV system using the I2NSF framework.

NSF facing interface between security controller and NSFs can be implemented by Network Configuration Protocol (NETCONF) [RFC6241] with a data modeling language called YANG [RFC6020] that describes function-level security services. A data model in [i2nsf-cap-interface-yang] can be used for the I2NSF capability interface, which is NSF facing interface.

The proposed framework of SDN-based security services can be combined to a security management architecture in [i2nsf-sec-mgnt-arch] for handling high-level security policies as well as low-level security policies.

Also, the proposed framework can enforce low-level security policies in NSFs by using a service function chaining (SFC) enabled I2NSF architecture in [i2nsf-sfc-enabled-arch].

5. Objectives

- Prompt reaction to new network attacks: SDN-based security services allow private networks to defend themselves against new sophisticated network attacks.

- Automatic defense from network attacks: SDN-based security services identify the category of network attack (e.g., malware and DDoS attacks) and take counteraction for the defense without the intervention of network administrators.

- Network-load-aware resource allocation: SDN-based security services measure the overhead of resources for security services and dynamically select resources considering load balance for the maximum network performance.

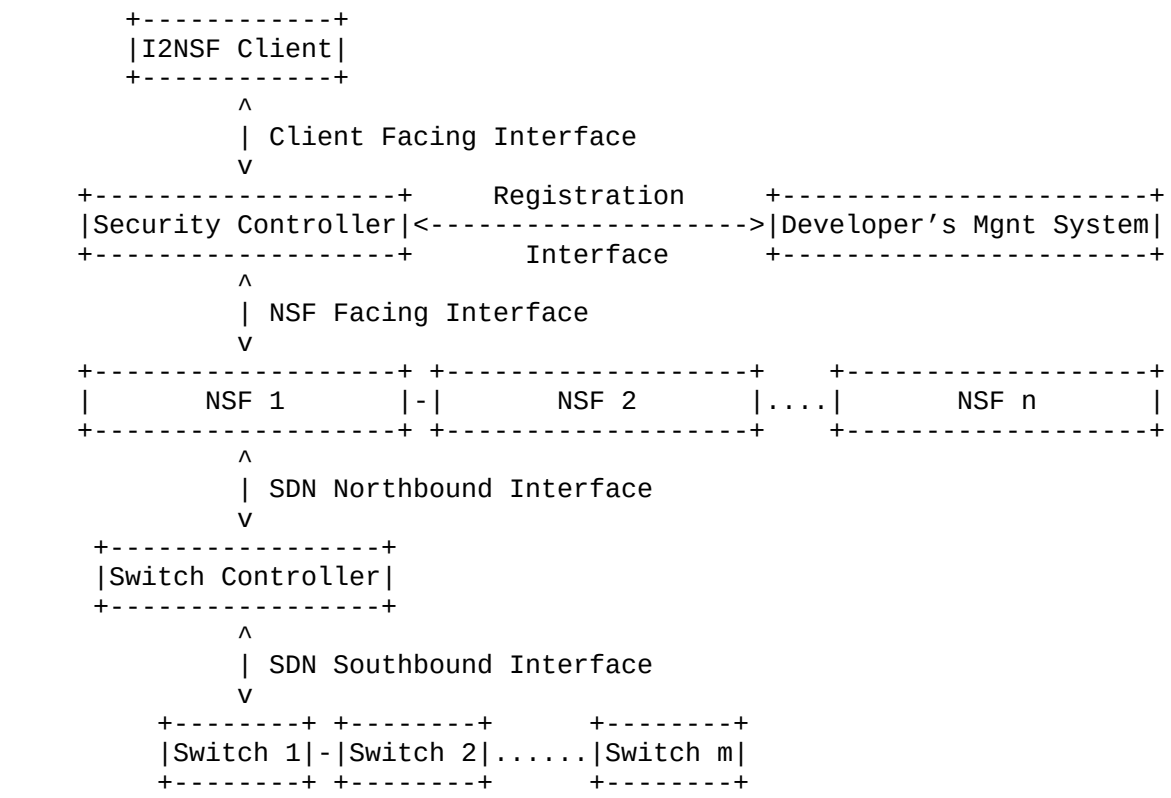


Figure 2: A Framework for SDN-based Security Services using I2NSF

6. Requirements

SDN-based security services provide dynamic and flexible network resource management to mitigate network attacks, such as malware and DDoS attacks. In order to support this capability, the requirements for SDN-based security services are described as follows:

- SDN-based security services are required to support the programmability of network resources to mitigate network attacks.
- SDN-based security services are required to support the orchestration of network resources and SDN applications to mitigate network attacks.
- SDN-based security services are required to provide an application interface allowing the management of access control policies in an autonomous and prompt manner.

- SDN-based security services are required to provide a resource-control interface for the control of network resources to mitigate network attacks.

- SDN-based security services are required to provide the logically centralized control of network resources to mitigate network attacks.

- SDN-based security services are required to support the seamless services to mitigate network attacks.

- SDN-based security services are required to provide the dynamic control of network resources to mitigate network attacks.

7. Use Cases

This section introduces three use cases for security services based on SDN: (i) centralized firewall system, (ii) centralized DDoS-attack mitigation system, and (iii) centralized VoIP/VoLTE security system.

7.1. Centralized Firewall System

For the centralized firewall system, a centralized network firewall can manage each network resource and firewall rules can be managed flexibly by a centralized server for firewall (called Firewall). The centralized network firewall controls each switch for the network resource management and the firewall rules can be added or deleted dynamically.

The procedure of firewall operations in the centralized firewall system is as follows:

1. Switch forwards an unknown flow’s packet to Switch Controller.

2. Switch Controller forwards the unknown flow’s packet to an appropriate security service application, such as Firewall.

3. Firewall analyzes the headers and contents of the packet.

4. If Firewall regards the packet as a malware’s packet with a suspicious pattern, it reports the malware’s packet to Switch Controller.

5. Switch Controller installs new rules (e.g., drop packets with the suspicious pattern) into switches.

6. The malware’s packets are dropped by switches.

For the above centralized firewall system, the existing SDN protocols can be used through standard interfaces between the firewall application and switches [RFC7149][ITU-T.Y.3300][ONF-OpenFlow] [ONF-SDN-Architecture].

Legacy firewalls have some challenges such as the expensive cost, performance, management of access control, establishment of policy, and packet-based access mechanism. The proposed framework can resolve these challenges through the above centralized firewall system based on SDN as follows:

- 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 network interfaces. Every network resource for firewall needs to check firewall rules according to network conditions. Firewalls can be adaptively deployed among network switches, depending on network conditions in the framework.

- The 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 firewall is a challenge. In the framework, firewall rules can be dynamically added for new malware.

- The establishment of policy: Policy should be established for each network resource. However, it is difficult to describe what flows are permitted or denied for firewall within a specific organization network under management. Thus, a centralized view is helpful to determine security policies for such a network.

- Packet-based access mechanism: Packet-based access mechanism is not enough for firewall in practice since the basic unit of access control is usually users or applications. Therefore, application level rules can be defined and added to the firewall system through the centralized server.

7.2. Centralized DDoS-attack Mitigation System

For the centralized DDoS-attack mitigation system, a centralized DDoS-attack mitigation can manage each network resource and manipulate rules to each switch through a centralized server for DDoS-attack mitigation (called DDoS-attack Mitigator). The centralized DDoS-attack mitigation system defends servers against

DDoS attacks outside private network, that is, from public network.

Servers are categorized into stateless servers (e.g., DNS servers) and stateful servers (e.g., web servers). For DDoS-attack mitigation, traffic flows in switches are dynamically configured by traffic flow forwarding path management according to the category of servers [AVANT-GUARD]. Such a managenent should consider the load balance among the switches for the defense against DDoS attacks.

The procedure of DDoS-attack mitigation operations in the centralized DDoS-attack mitigation system is as follows:

1. Switch periodically reports an inter-arrival pattern of a flow's packets to Switch Controller.

2. Switch Controller forwards the flow's inter-arrival pattern to an appropriate security service application, such as DDoS-attack Mitigator.

3. DDoS-attack Mitigator analyzes the reported pattern for the flow.

4. If DDoS-attack Mitigator regards the pattern as a DDoS attack, it computes a packet dropping probability corresponding to suspiciousness level and reports this DDoS-attack flow to Switch Controller.

5. Switch Controller installs new rules into switches (e.g., forward packets with the suspicious inter-arrival pattern with a dropping probability).

6. The suspicious flow's packets are randomly dropped by switches with the dropping probability.

For the above centralized DDoS-attack mitigation system, the existing SDN protocols can be used through standard interfaces between the DDoS-attack mitigator application and switches [RFC7149] [ITU-T.Y.3300][ONF-OpenFlow][ONF-SDN-Architecture].

The centralized DDoS-attack mitigation system has challenges similar to the centralized firewall system. The proposed framework can resolve these challenges through the above centralized DDoS-attack mitigation system based on SDN as follows:

- Cost: The cost of adding DDoS-attack mitigators to network resources such as routers, gateways, and switches is substantial due to the reason that we need to add DDoS-attack mitigator on each network resource. To solve this, each network resource can be managed centrally such that a single DDoS-attack mitigator is

manipulated by a centralized server.

- Performance: The performance of DDoS-attack mitigators is often slower than the link speed of network interfaces. The checking of DDoS attacks may reduce the performance of the network interfaces. DDoS-attack mitigators can be adaptively deployed among network switches, depending on network conditions in the framework.

- The management of network resources: Since there may be hundreds of network resources in an administered network, the dynamic management of network resources for performance (e.g., load balancing) is a challenge for DDoS-attack mitigation. In the framework, as dynamic network resource management, traffic flow forwarding path management can handle the load balancing of network switches [AVANT-GUARD]. With this management, the current and near-future workload can be spread among the network switches for DDoS-attack mitigation. In addition, DDoS-attack mitigation rules can be dynamically added for new DDoS attacks.

- The establishment of policy: Policy should be established for each network resource. However, it is difficult to describe what flows are permitted or denied for new DDoS-attacks (e.g., DNS reflection attack) within a specific organization network under management. Thus, a centralized view is helpful to determine security policies for such a network.

7.3. Centralized VoIP/VoLTE Security System

For the centralized VoIP/VoLTE security system, a centralized VoIP/VoLTE security system can monitor each VoIP/VoLTE flow and manage VoIP/VoLTE security rules controlled by a centralized server for VoIP/VoLTE security service (called VoIP IPS). The VoIP/VoLTE security system controls each switch for the VoIP/VoLTE call flow management by manipulating the rules that can be added, deleted or modified dynamically.

The procedure of VoIP/VoLTE security operations in the centralized VoIP/VoLTE security system is as follows:

1. A switch forwards an unknown call flow’s signal packet (e.g., SIP packet) to Switch Controller. Also, if the packet belongs to a matched flow’s packet related to SIP (called matched SIP packet), Switch forwards the packet to Switch Controller so that the packet can be checked by an NSF for VoIP (i.e., VoIP IPS) via Switch Controller, which monitors the behavior of its SIP call.

2. Switch Controller forwards the unknown flow’s packet or the matched SIP packet to an appropriate security service function,

such as VoIP IPS.

3. VoIP IPS analyzes the headers and contents of the signal packet, such as IP address, calling number, and session description [RFC4566].

4. If VoIP IPS regards the packet as a spoofed packet by hackers or a scanning packet searching for VoIP/VoLTE devices, it requests the Switch Controller to block that packet and the subsequent packets that have the same call-id.

5. Switch Controller installs new rules (e.g., drop packets) into switches.

6. The illegal packets are dropped by switches.

For the above centralized VoIP/VoLTE security system, the existing SDN protocols can be used through standard interfaces between the VoIP IPS application and switches [RFC7149][ITU-T.Y.3300] [ONF-OpenFlow][ONF-SDN-Architecture].

Legacy hardware based VoIP IPSes have some challenges, such as provisioning time, the granularity of security, expensive cost, and the establishment of policy. The proposed framework can resolve these challenges through the above centralized VoIP/VoLTE security system based on SDN as follows:

- Provisioning: The provisioning time of setting up a legacy VoIP IPS to network is substantial because it takes from some hours to some days. By managing the network resources centrally, VoIP IPS can provide more agility in provisioning both virtual and physical network resources from a central location.

- The granularity of security: The security rules of a legacy VoIP IPS are compounded considering the granularity of security. The proposed framework can provide more granular security by centralizing security control into a switch controller. The VoIP IPS can effectively manage security rules throughout the network.

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

- The establishment of policy: Policy should be established for each network resource. However, it is difficult to describe what flows are permitted or denied for VoIP IPS within a specific

organization network under management. Thus, a centralized view is helpful to determine security policies for such a network.

So far this document has described the procedure and impact of the three use cases for security services. To support these use cases in the proposed framework, a data model described in [i2nsf-cap-interface-yang] can be used as NSF facing interface along with NETCONF [RFC6241].

8. Security Considerations

The proposed SDN-based framework in this document is derived from the I2NSF framework [i2nsf-framework], so the security considerations of the I2NSF framework should be included in this document. Therefore, proper secure communication channels should be used the delivery of control or management messages among the components in the proposed framework.

This document shares all the security issues of SDN that are specified in the "Security Considerations" section of [ITU-T.Y.3300].

9. Acknowledgements

This document was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) [10041244, Smart TV 2.0 Software Platform] and by MSIP/IITP [R0166-15-1041, Standard Development of Network Security based SDN].

This document has greatly benefited from inputs by Jinyong Kim, Daeyoung Hyun, Mahdi Daghmehchi-Firoozjaei, and Geumhwan Cho.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[i2nsf-framework] Lopez, E., Lopez, D., Dunbar, L., Strassner, J., Zhuang, X., Parrott, J., Krishnan, R., and S. Durbha, "Framework for Interface to Network Security Functions", draft-ietf-i2nsf-framework-01, June 2016.

[RFC6241] Enns, R., Bjorklund, M., Schoenwaelder, J., and A. Bierman, "Network Configuration Protocol (NETCONF)", RFC 6241, June 2011.

[RFC6020] Bjorklund, M., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, October 2010.

10.2. Informative References

[i2nsf-cap-interface-yang] Jeong, J., Kim, J., Hyun, D., Park, J., and T. Ahn, "YANG Data Model of Interface to Network Security Functions Capability Interface", draft-jeong-i2nsf-capability-interface-yang-00, July 2016.

[i2nsf-sec-mgnt-arch] Kim, H., Ko, H., Oh, S., Jeong, J., and S. Lee, "An Architecture for Security Management in I2NSF Framework", draft-kim-i2nsf-security-management-architecture-01, July 2016.

[i2nsf-sfc-enabled-arch] Hyun, S., Woo, S., Yeo, Y., Jeong, J., and J. Park, "Service Function Chaining-Enabled I2NSF Architecture", draft-hyun-i2nsf-sfc-enabled-i2nsf-00, July 2016.

[RFC7149] Boucadair, M. and C. Jacquenet, "Software-Defined Networking: A Perspective from within a Service Provider Environment", RFC 7149, March 2014.

[ITU-T.Y.3300] Recommendation ITU-T Y.3300, "Framework of Software-Defined Networking", June 2014.

[ONF-OpenFlow] ONF, "OpenFlow Switch Specification (Version 1.4.0)", October 2013.

[ONF-SDN-Architecture] ONF, "SDN Architecture", June 2014.

[ITU-T.X.1252] Recommendation ITU-T X.1252, "Baseline Identity Management Terms and Definitions", April 2010.

[ITU-T.X.800] Recommendation ITU-T X.800, "Security Architecture for Open Systems Interconnection for CCITT Applications", March 1991.

[AVANT-GUARD] Shin, S., Yegneswaran, V., Porras, P., and G. Gu, "AVANT-GUARD: Scalable and Vigilant Switch Flow Management in Software-Defined Networks", ACM CCS, November 2013.

[ETSI-NFV] ETSI GS NFV 002 V1.1.1, "Network Functions Virtualisation (NFV); Architectural Framework", October 2013.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, July 2006.

Appendix A. Changes from draft-jeong-i2nsf-sdn-security-services-04

The following changes were made from draft-jeong-i2nsf-sdn-security-services-04:

- According to the change of terminology in the I2NSF framework, the names of the components and interfaces are updated as follows: Application Controller -> I2NSF Client, Security Function (SF) -> Network Security Function (NSF), Vendor System -> Developer's Management System, Service Layer Interface -> Client Facing Interface, Capability Layer Interface -> NSF Facing Interface.

- Three use cases described in this document can use a data model corresponding to the information model for the I2NSF capability interface.

- The proposed framework of SDN-based security services can be combined to a security management architecture for handling security policies.

- The proposed framework can enforce low-level security policies in NSFs by using a service function chaining (SFC) enabled I2NSF architecture.

Authors' Addresses

Jaehoon Paul Jeong
Department of Software
Sungkyunkwan University
2066 Seobu-Ro, Jangan-Gu
Suwon, Gyeonggi-Do 16419
Republic of Korea

Phone: +82 31 299 4957
Fax: +82 31 290 7996
EMail: pauljeong@skku.edu
URI: http://iotlab.skku.edu/people-jaehoon-jeong.php

Hyoungshick Kim
Department of Software
Sungkyunkwan University
2066 Seobu-Ro, Jangan-Gu
Suwon, Gyeonggi-Do 16419
Republic of Korea

Phone: +82 31 299 4324
Fax: +82 31 290 7996
EMail: hyoung@skku.edu
URI: http://seclab.skku.edu/people/hyoungshick-kim/

Jung-Soo Park
Electronics and Telecommunications Research Institute
218 Gajeong-Ro, Yuseong-Gu
Daejeon 305-700
Republic of Korea

Phone: +82 42 860 6514
EMail: pjs@etri.re.kr

Tae-Jin Ahn
Korea Telecom
70 Yuseong-Ro, Yuseong-Gu
Daejeon 305-811
Republic of Korea

Phone: +82 42 870 8409
EMail: taejin.ahn@kt.com

Se-Hui Lee
Korea Telecom
70 Yuseong-Ro, Yuseong-Gu
Daejeon 305-811
Republic of Korea

Phone: +82 42 870 8162
EMail: sehuilee@kt.com

Network Working Group A. Pastor
Internet-Draft D. Lopez
Intended status: Experimental Telefonica I+D
Expires: April 28, 2015 October 25, 2014

Access Use Cases for an Open OAM Interface to Virtualized Security Services
draft-pastor-i2nsf-access-usecases-00

Abstract

This document describes the use cases for providing network security as a service in the access network environment. It considers both mobile and residential access.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 28, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 3
2. Requirements Language 4
3. Actors in the Access Environment 4
4. Operator-Managed Security Functions 4
4.1. vNSF Deployment 5
4.2. vNSF Customer Provisioning 5
5. Customer-Managed Security Functions 5
5.1. Self-Provisioning 5
5.2. Validation 5
6. Policies and Configuration 6
7. Security Functions at the Access Network 7
7.1. Traffic Inspection 7
7.2. Traffic Manipulation 7
7.3. Impersonation 7
8. Security Considerations 8
9. IANA Considerations 8
10. References 8
10.1. Normative References 8
10.2. Informative References 8
Authors' Addresses 8

1. Introduction

This document describes the use cases for an open OAM interfce to virtualized network security services in residential and mobile network access.

Not only enterprise customers, but also residential and mobile ones are becoming more and more aware of the need for security, just to find that security services are hard to operate and become expensive in the case of reasonably sophisticated ones. This general trend has caused that numerous operators and security vendors start to leverage cloud-based models to deliver security solutions. In particular, the methods around Network Function Virtualization (NFV) are meant to facilitate the management of various resources for the benefit of customers, who may not own or physically host those network functions.

This document analyzes the use cases for the provision, operation and management of virtualized Network Security Function (vNSF) in the access network environment, as shown in the following figure.

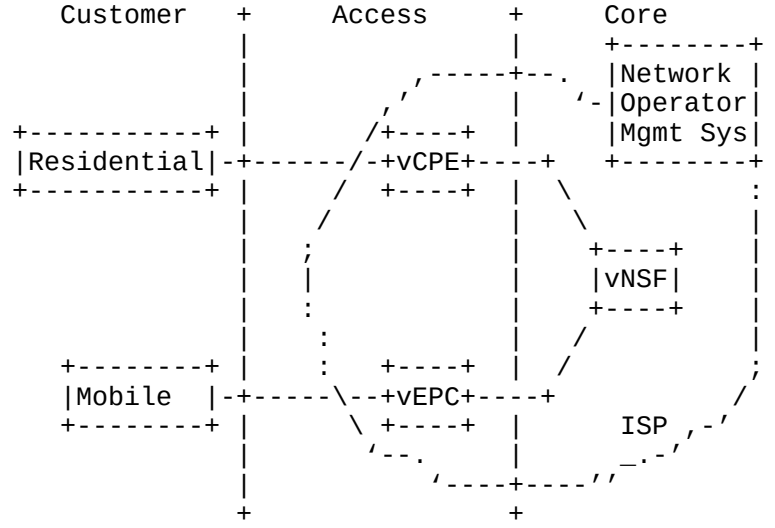


Figure 1: Customer Access Network

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

In this document, these words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying RFC-2119 significance.

3. Actors in the Access Environment

Different types of actors can use an open OAM interface to the vNSFs to allocate and operate network security functions. The envisioned actors are:

- Network operators that provide and manage vNSF in their administrative domain or through external providers.
- Customers, accessing through the Network Operator, and requiring a security service implemented by one or more vNSF.

The access network technology environment also defines the characteristics of the type of access in each use case:

- Closed environments where there is only one administrative network domain. More permissive access controls and lighter validation shall be allowed inside the domain because of the protected environment. Integration with existing identity management systems is also possible.
- Open environments where some vNSFs can be hosted in external administrative domains, and more restrictive security controls are required. The interfaces to the vNSFs must use trusted channels. Identity frameworks and federations are common models for authentication and Authorization.

4. Operator-Managed Security Functions

The Virtual CPE described in [NFVUC] use cases #5 and #7 cover the model of virtualization for mobile and residential access, where the operator may offload security services from the customer local environment (or even the terminal) to the operator infrastructure supporting the access network.

This use case defines the operator interaction with vNSF through

automated interfaces, typically by B2B communications performed by the operator management systems (OSS/BSS).

4.1. vNSF Deployment

The deployment process consists of instantiating a vNSF on a Virtualization Infrastructure (NFVI), within the operator administrative domain(s) or an external domain. This is a required step before a customer can subscribe to a security service supported in the vNSF.

4.2. vNSF Customer Provisioning

Once a vNSF is deployed, any customer can subscribe to it. The provisioning lifecycle includes:

- Customer enrollment and cancellation of the subscription to a vNSF.
- Configuration of the vNSF, based on specific configurations or derived from common security policies defined by the operator.
- Retrieve and list of the vNSF functionalities, extracted from a manifest or a descriptor. The network operator management systems can demand this information to offer detailed information through the commercial channels to the customer.

5. Customer-Managed Security Functions

This is an alternative use case where the management is delegated directly to the customer. The open OAM interface permits direct interactions between the vNSF and the customer. This allows customers to have dynamic and flexible interactions with security services, more adequate for dynamic allocation of these virtualized security services.

5.1. Self-Provisioning

This process allows a residential or mobile customer to enroll on its own to a security service provided by a vNSF or a set of vNSF. The open OAM interface must support the enrollment process.

5.2. Validation

Customers MAY require to validate vNSF availability, provenance, and its correct execution. The validation process includes at least:

- o Integrity of the vNSF. The vNSF is not manipulated.
- o Isolation. The execution of the vNSF is self-contained for privacy requirements in multi-tenancy scenarios.

6. Policies and Configuration

vNSF configurations can vary from simple rules (i.e. block a DDoS attack) to very complex configuration (i.e. define a user firewall rules per application, protocol, source and destination port and address). The possibility of using configuration templates per vNSF type is a common option as well.

The operator can push security policies using complex configurations in their managed vNSF through its management system. The open OAM interface has to accommodate this application-driven behavior.

Computer-savvy customers may pursue a similar application-driven configuration through the open OAM interface, but standard residential and mobile customers may prefer to use the definition of security policies in the form of close-to-natural-language sentences with high-level directives or a guide configuration process. The representation for these policies will be of the form:

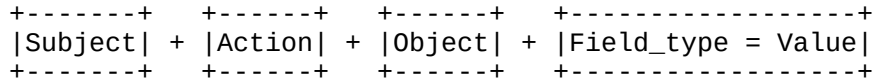


Figure 2: High-Level Security Policy Format

Subject indicates the customer or device in the access.

Action can include a variety of actions: check, redirect, allow, block, record, inspect...

Object can be optional and specifies the nature of the action. The default is all the customer traffic, but others possible values are connections and connections attempts.

Field_type allows to create fine-grained policies, including destinations list (i.e. IPs, domains), content types (i.e. files, emails), windows of time (i.e. weekend), protocol or network service (i.e. HTTP).

An example of a customer policy is:

"My son is allowed to access Facebook from 18:30 to 20:00"

7. Security Functions at the Access Network

This section collects a representative list of use cases of possible vNSFs that requires an open OAM interface for control and management.

7.1. Traffic Inspection

A common use case for customers accessing the Internet or additional services through it is security supervision. Some examples are:

- Intrusion detection systems
- Deep packet inspection
- Data leakage protection

An open OAM interface will allow the configuration of the vNSF inspection features: signatures updates, behavioral parameters or type of traffic to supervise.

7.2. Traffic Manipulation

A more intrusive use case of vNSF includes the capacity of manipulate the traffic at the access network segment. Some examples are:

- Redirect traffic, as in the case of captive portals
- Block traffic: Firewalls, intrusion prevention system, anti-DoS mechanisms...
- Encrypt traffic: VPN services that encapsulate and encrypt the user traffic. A SSL VPN is a representative example.

An open OAM interface will allow the configuration of the vNSF manipulation features, such as redirect and block rules.

7.3. Impersonation

Some vNSFs can impersonate a customer service or Internet service to provide security functions. Some examples are:

- Honeypots, impersonating customer services, such as HTTP, NetBios or SSH

- Anonymization services, hiding the source identity, as in the case of TOR

An open OAM interface will allow the configuration of the vNSF impersonation features, like the service to impersonate.

8. Security Considerations

TBD

9. IANA Considerations

This document requires no IANA actions.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

10.2. Informative References

[NFVUC] "ETSI NFV Group Specification, Network Functions Virtualization (NFV) Use Cases", <http://www.etsi.org/deliver/etsi_gs/NFV/001_099/001/01.01.01_60/gs_NFV001v010101p.pdf>.

Authors' Addresses

Antonio Pastor
Telefonica I+D
Don Ramon de la Cruz, 82
Madrid, 28006
Spain

Phone: +34 913 128 778
Email: antonio.pastorperales@telefonica.com

Diego R. Lopez
Telefonica I+D
Don Ramon de la Cruz, 82
Madrid, 28006
Spain

Phone: +34 913 129 041
Email: diego.r.lopez@telefonica.com

Internet Engineering Task Force K. Wang
Internet-Draft X. Zhuang
Intended status: Informational China Mobile
Expires: September 7, 2015 March 6, 2015

Integrated Security with Access Network Use Case
draft-qi-i2nsf-access-network-usecase-02

Abstract

In traditional telecommunication system, operators usually provide general and limited security protection service for users during access (e.g. AKA in 3G/4G network). Now, with the development of network virtualization technology and data center, physical network devices can be replaced by network function softwares which are running on virtual machines and the network function can be flexible and elastic. Operators can provide users with more security services. So this interfaces between operator's network and users are highly desired. These interfaces will be used to request/achieve (Virtual) Network Security Functions from operator's network. This draft describes use cases for using the interface in operator's network environment.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 31, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
2. Conventions used in this document 3
3. Use case summary 3
4. Use case for Instantiation and Configuration of Security Service Function 5
5. Use case for Updating Security Service Function 5
6. Use case for Collecting and Feedback of Status of Security Service Function 5
7. The Benefits 6
8. IANA Considerations 6
9. Informative References 6
Authors' Addresses 6

1. Introduction

This draft is a revised version of draft-qi-i2nsf-access-network-usecase and refines the original use cases. In draft-qi-i2nsf-access-network-usecase, an interface between UE and network was described while this draft describes two interfaces.Users can use client to achieve security service of operator via these interfaces. The user can be an enterprise, an enterprise user, administrator of operator and so on. The revisions details as below: 1.For original use case-Interface about sending security configuration information from network to UE: All examples have been deleted and network did not send configuration information to UE via interface. Instead Users will send security service requests to security controller to configure NSF(s). 2.For original use case-Interface about optional security function negotiation between Network and UE: All examples have been deleted and there is no security function negotiation between network and UE. Instead Users will send security service request to security controller to configure NSF(s). 3.For original

use case-UE proposed security request to the network: The original interactions between user and network will be more concrete. For example, the original interaction between user and specific network element will be revised into interaction between user's client and security controller. The interaction between specific network element and security function settings will be described in detail. 4.For original section of Abstraction and The Benefits: Corresponding modifications have been made to match revised use cases better.

2. Conventions used in this document

The section clarifies the intended meaning of specific terms used within this document.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

In this document, these words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying [RFC2119] significance.

3. Use case summary

This draft describes use cases of users (e.g. enterprise user, operator's administrator and so on) using operators' flexible security services. For example, a user can request a security service through a client (e.g. APP, BSS/OSS, OAM etc.).An operator's network entity(e.g. gateway) can invoke (v)NSF(s) according to user's service request. In order to make the description more clear, we call operator's network entity as security controller. The interaction between entities above(i.e. client, security controller, NSF) can be showed as below:

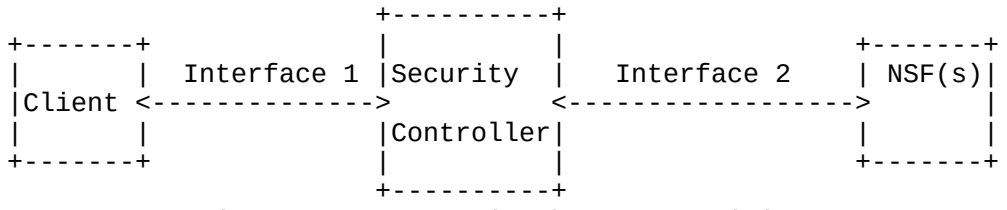


Figure 1. Interaction between Entities

Interface 1 is used for receiving security requirements from client and translating them into commands that NSF(s) can understand and execute. Moreover, it is also responsible for giving feedback of NSFs' security statistics to client.Interface 2 is used for interacting with NSFs according to commands. Moreover, it is also

responsible for receiving the results of commands from NSFs.NSF mentioned in this draft includes virtualized NSF and physical NSF.

4. Use case for Instantiation and Configuration of Security Service Function

Client sends collected security requirements through interface 1 to the security controller in operator's network which then translates them into a security function or a set of security functions then the corresponding NSFs are instantiated and configured through interface 2.For example, an enterprise user A is a tenant of operator data center and wants to filter all TCP data packets flowing to A's network. Such a requirement is sent from client to security controller through interface 1. The security controller translates the requirement into a firewall function and then instantiates a firewall NSF through interface 2. The corresponding filter rule is also configured onto this firewall NSF.

5. Use case for Updating Security Service Function

User can use client to update security service function, including adding/deleting a security service function and updating configurations at former security service function. For example,a user who has instantiated a security service before wants to enable an IDS service additionally, this requirement will be sent to security controller through interface 1 and be translated and then security controller instantiates and configures an IDS NSF through interface 2. Another example is that if the user A mentioned in use case 1 wants to filter all UDP packets besides TCP packets, client sends this requirement to security controller through interface 1 and then security controller configures translated requirement onto the former firewall NSF.

6. Use case for Collecting and Feedback of Status of Security Service Function

When users want to get the executing status of security service, they can request the status statistics information of NSF(s) from client. Security controller can collect NSFs' status statistics information through interface 2 and give feedback to client through interface 1, which is helpful for user analyzing or updating security requirements. Users can collect status statistics information of NSF(s) related to their security service and can also be authorized to collect all NSFs' status statistics information for the analysis of big data for network security like the overall security status of the network in operator's data center.

7. The Benefits

There are numerous benefits by defining such interfaces. Operators could provide more flexible and customized security services for specific users and this would provide more efficient and secure protection to each user.

8. IANA Considerations

TBD

9. Informative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

Authors' Addresses

Ke Wang
China Mobile
32 Xuanwumenxi Ave,Xicheng District
Beijing 100053
China

Email: wangkeyj@chinamobile.com

Xiaojun Zhuang
China Mobile
32 Xuanwumenxi Ave, Xicheng District
Beijing 100053
China

Email: zhuangxiaojun@chinamobile.com

Minpeng Qi
China Mobile
32 Xuanwumenxi Ave,Xicheng District
Beijing 100053
China

Email: qiminpeng@chinamobile.com

Network Working Group
Internet Draft
Intended Status: Informational

M. Zarny
Goldman Sachs
S. Magee
F5
N. Leymann
Deutsche Telecom
L. Dunbar
Huawei

Expires: April 28, 2015

October 25, 2014

I2NSF Data Center Use Cases
draft-zarny-i2nsf-data-center-use-cases-00

Abstract

This document describes data center use cases and their requirements that a common Interface to Network Security Functions (I2NSF) needs to take into account.

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/1id-abstracts.html

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

Copyright and License Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 3
2. Conventions used in this document 3
3. Terminology 4
4. On-demand, elastic deployment of firewalls 4
4.1 On demand virtual firewall deployment in cloud data centers 5
5. Firewall policy deployment automation 6
5.1 Client-specific security policy in cloud VPNs 6
6. Key requirements for the use cases 6
7. Conclusion 7
8. Security considerations 8
9. IANA considerations 8
10. References 8
10.1 Normative references 8
10.2 Informative references 8
11. Acknowledgments 8
12. Authors' addresses 8

1. Introduction

Enterprises today increasingly consume cloud-based network security functions. The reasons are the same as those for the move toward cloud computing: greater economies of scale; faster service delivery; greater flexibility to respond to changing requirements; faster deployment of more sophisticated solutions; among others.

The cloud security services can in theory be offered in a number of ways. They can be operated by service providers or enterprises themselves; they can be run on shared or dedicated infrastructure; they can be deployed off- or on-premises; or any combination thereof. In practice, however, since most firms today possess neither the expertise nor resources to build and manage clouds, most firms that consume cloud-based security services do so on off-premise provider-managed clouds.

In response, providers and security vendors offer cloud-based models to deliver security solutions. Providers in particular are striving to standardize the offering methodologies through efforts like Network Functions Virtualization (NFV).

I2NSF is an IETF effort to standardize the interface for network security functions offered on any kind of cloud regardless of its location or operator. Since the term "network security service" can mean many things, we will limit the term to include only the following services in this draft.

* Firewall

* DDOS/Anti-DOS (Distributed Denial-of-Service/Anti-Denial-of-Service)

* AAA (Authentication, Authorization, Accounting)

* Remote identity management

* Secure key management

* IDS/IPS (Intrusion Detection System/Intrusion Prevention System)

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

3. Terminology

Cloud-scale network resources: Networked resources which provide various functions (related to infrastructure, platform, software, etc.) in a scalable, automated and secure fashion. Resources in the cloud may be fully owned, operated, and used by a single organization; dedicated to a single client and managed by a provider; shared amongst several clients; hosted on-premises or off-premises of an organization; or a combination thereof. In the context of this draft, a cloud offers network security services.

DC: Data Center

Domain relationships: The term "Domain" in this draft has different connotations in different scenarios:

Client <-> Provider relationship, i.e. a client requesting network service functions from its provider;

Domain A <-> Domain B relationship, i.e. one operator domain requesting network service functions from another operator domain; or

Applications <-> Network relationship, an application (e.g., cluster of servers) requesting some functions from network.

Network function: In the context of I2NSF, the term "network function" describes services that provide network functions including L4-L7 functions. The network service functions may not necessarily be owned or hosted by consumers of those functions. Furthermore, the network functions may be hosted on physical appliances, inside containers, or inside VMs instantiated on common compute servers (e.g., the ETSI NFV defined Virtualized Network Functions).

Virtual Security Function: A security function that can be requested by one domain but may be owned or managed by another domain.

Cloud-based security functions: Used interchangeably with the "Virtual Security Functions" in this draft.

4. On-demand, elastic deployment of firewalls

Network security devices such as firewalls may need to be added or removed dynamically for a number of reasons. It may have been explicitly requested by the user, or triggered by a pre-agreed-upon service level agreement (SLA) between the user and the provider of

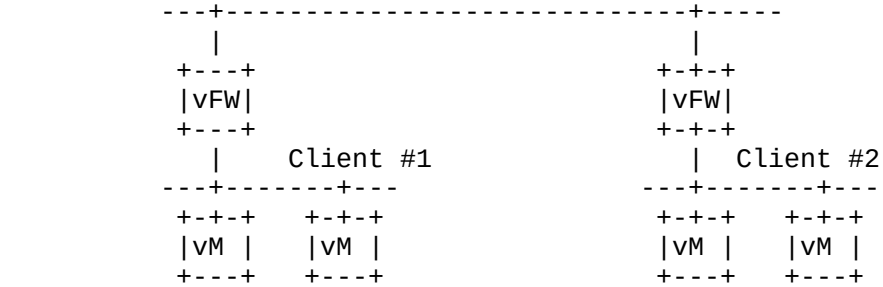
the service. For example, the service provider may be required to add more firewall capacity within a set timeframe whenever the bandwidth utilization hits a certain threshold for a specified period. This capacity expansion could result in adding new instances of firewalls. Likewise, a service provider may need to provision a new firewall instance in a completely new environment due to a new requirement.

The on-demand, dynamic nature of deployment essentially requires that the network security "devices" be in software or virtual form factors, rather than in a physical appliance form. (This is a provider-side concern. Users of the firewall service are agnostic, as they should, as to whether or not the firewall service is run on a VM or any other form factor. Indeed, they may not even be aware that their traffic traverses firewalls.)

Furthermore, new firewall instances need to be placed in the "right zone" (domain). The issue applies not only to multi-tenant environments where getting the tenant right is of paramount importance but also to environments owned and operated by a single organization with its own service segregation policies. For example, an enterprise may mandate that firewalls serving Internet traffic and business-to-business (B2B) traffic be separate; or that IPS/IDS services for investment banking and non-banking traffic be separate for regulatory reasons.

4.1 On demand virtual firewall deployment in cloud data centers

A service provider operated cloud data center could serve tens of thousands of clients. Clients' compute servers are typically hosted on virtual machines (VMs), which could be deployed across different server racks located in different parts of the data center. It is often not technically and/or financially feasible to deploy dedicated physical firewalls to suit each client's myriad security policy requirements. What is needed is the ability to dynamically deploy virtual firewalls for each client's set of servers based on established security policies and underlying network topologies.



5. Firewall policy deployment automation

Firewall configuration today is a highly complex process that involves consulting established security policies, translating those policies into firewall rules, further translating those rules into vendor-specific configuration sets, identifying all the firewalls, and pushing configurations to those firewalls.

This is often a time consuming, complex and error-prone process even within a single organization/enterprise framework. It becomes far more complex in provider-owned cloud networks that serve myriad customers.

Automation can help address many of these issues. Automation works best when it can leverage a common set of standards that will work across multiple entities.

5.1 Client-specific security policy in cloud VPNs

Clients of service provider operated cloud data centers need not only secure virtual private networks (VPNs) but also virtual security functions that enforce the clients' security policies. The security policies may govern communications within the clients' own virtual networks and those with external networks. For example, VPN service providers may need to provide firewall and other security services to their VPN clients. Today, it is generally not possible for clients to dynamically view, much less change, what, where and how security policies are implemented on their provider-operated clouds. Indeed, no standards-based framework that allows clients to retrieve/manage security policies in a consistent manner across different providers exists.

6. Key requirements for the use cases

The I2NSF framework should provide a set of standard interfaces that facilitate:

* Dynamic creation, enablement, disablement, and removal of network security applications;

* Policy-driven placement of new service instances in the right administrative domain;

* Attachment of appropriate security and traffic policies to the service instances

* Management of deployed instances in terms of fault monitoring, utilization monitoring, event logging, inventory, etc.

Moreover, an I2NSF must support different deployment scenarios:

* Single and multi-tenant environments: The term multi-tenant does not mean just different companies subscribing to a provider's cloud offering. It can for instance cover administrative domains/departments within a single firm that require different security and traffic policies.

* Premise-agnostic: Said network security services may be deployed on premises or off premises of an organization.

The I2NSF framework should provide a standard set of interfaces that enable:

* Translation of security policies into functional tasks. Security policies may be carried out by one or more security service functions. For example, a security policy may be translated into an IDS/IPS policy and a firewall policy for a given application type.

* Translation of functional tasks into vendor-specific configuration sets. For example, a firewall policy needs to be converted to vendor-specific configurations.

* Retrieval of information such as configuration, utilization, status, etc. Such information may be used for monitoring, auditing, troubleshooting purposes. The above functions should be available in single- or multi-tenant environments as well as on-premise or off-premise clouds.

7. Conclusion

The need for common interfaces to network service functions goes beyond network security functions described here. Efforts like NFV will drive efforts to address this broad need. This draft covers common network security functions deployed in data centers as a way to scope the problem set. The use cases here are relevant to service provider and large enterprise networks, and they can all benefit significantly from an I2NSF.

We recommend the IETF to start a program to establish a common framework for network security functions that will address the issues raised here.

8. Security considerations

TBD.

9. IANA considerations

This document requires no IANA actions. RFC Editor: Please remove this section before publication.

10. References

10.1 Normative references

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC7297] Boucadair, M., "IP Connectivity Provisioning Profile", RFC7297, April 2014.

10.2 Informative references

[PS] Dunbar, et al, "Dynamic Network Security as a Service Problem Statement", <draft-dunbar-nsaas-problem-statement-00>, July 2014.

[GS-NFV] ETSI NFV Group Specification, Network Functions Virtualizsation (NFV) Use Cases. ETSI GS NFV 001v1.1.1, 2013.

[Boucadair-framework] Boucadair, M., et al, "Differentiated Service Function Chaining Framework", <draft-boucadair-service-chaining-framework-00>; Aug 2013

[SC-MobileNetwork] Haeffner, W. and N. Leymann, "Network Based Services in Mobile Network", IETF87 Berlin, July 29, 2013

[Application-SDN] Giacomonni, J., "Application Layer SDN", Layer 123 ONF Presentation, Singapore, June 2013

11. Acknowledgments

We would like to acknowledge Andrew Malis for his review and contribution.

12. Authors' addresses

Myo Zarny
Goldman Sachs
Email: myo.zarny@gs.com

Sumandra Majee
F5 Netowrks
Email: lal2ghar@gmail.com

Nic Leymann
Deutsche Telekom
Email: n.leymann@telekom.de

Linda Dunbar
Huawei
Email: linda.dunbar@huawei.com

Network Working Group
INTERNET-DRAFT
Intended Status: Informational

S. Hares
Huawei
D. Zhang

H. Moskowitz
HTT Consulting
H.Rafiee
Rozanak

Expires: January 6, 2016

July 6, 2015

Analysis of Existing Work for I2NSF
<draft-zhang-gap-analysis-06.txt>

Abstract

This document analyzes the status of the arts in industries and the existing IETF work/protocols that are relevant to I2NSF. existing IETF work/protocols that are relevant to the Interface to Network Security Function (I2NSF). The I2NSF focus is to define data models and interfaces in order to control and monitor the physical and virtual aspects of network security functions.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 6, 2016.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved. This document is subject to

BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 4
1.1. What is I2NSF 4
1.2. Structure of this Document 5
1.3. Terms and Definitions 5
1.3.1. Requirements Terminology 5
1.3.2. Definitions 5
2. IETF Gap analysis 7
2.1. Traffic Filters 7
2.1.1. Overview 7
2.1.1.1. Data Flow Filters in NETMOD and I2RS 7
2.1.1.2. I2NSF Gap analysis 9
2.1.2. Middle-box Filters 9
2.1.2.1. Midcom 9
2.1.3. Security Work 10
2.1.3.1. Overview 10
2.1.3.2. Security Work and Filters 11
2.1.3.3. I2NSF interaction 11
2.1.3.4. Benefits from the Interaction 12
3. ETSI NFV 13
3.1. ETSI Overview 13
3.2. I2NSF Gap Analysis 14
4. OPNFV 15
4.1. OPNFV Moon Project 15
4.2. Gap Analysis for OPNFV Moon Project 16
5. OpenStack Security Firewall 16
5.1. Overview of API for Security Group 17
5.2. Overview of Firewalls as a Service 17
5.3. I2NSF Gap analysis 18
6. CSA Secure Cloud 18
6.1. CSA Overview 18
6.1.1. CSA Security as a Service(SaaS) 19
6.1.2. Identity Access Management (IAM) 20
6.1.3. Data Loss Prevention (DLP) 20
6.1.4. Web security(Web)) 21
6.1.5. Email Security (email)) 22
6.1.6. Security Assessment 24
6.1.7. Intrusion Detection 24
6.1.8. Security Information and Event Management(SEIM) . . 25
6.1.9. Encryption 26
6.1.10. Business Continuity and Disaster Recovery (BC/DR) . 27

6.1.11. Network Security Devices 28
6.2. I2NSF Gap Analysis 29
7. In-depth Review of IETF protocols 29
7.1. NETCONF and RESTCONF 29
7.2. I2RS Protocol 30
7.3. NETMOD Yang modules 31
7.4. COPS 31
7.5. PCP 32
7.6. NSIS - Next steps in Signalling 33
8. Security Considerations 34
9. IANA Considerations 34
10. References 34
10.1. Normative 34
10.2. Informative 34
Authors' Addresses 42

1. Introduction

This documents provides a gap analysis for I2NSF.

1.1. What is I2NSF

The Network Security Function (NSF) in a network ensures integrity, confidentiality and availability of network communications, detects unwanted activity, and blocks out or at least mitigates the effects of unwanted activity. NSF devices are provided and consumed in increasingly diverse environments. For example, users of NSFs could consume network security services offered on multiple security products hosted one or more service provider,their own enterprises, or a combination of the two.

The lack of standard interfaces to control and monitor the behaviour of NSFs, makes it virtually impossible for security service providers to automate service offerings that utilize different security functions from multiple vendors.

The Interface to NSF devices (I2NSF) work proposes to standardize a set of software interfaces and data modules to control and monitor the physical and virtual NSFs. Since different security vendors support different features and functions, the I2NSF will focus on the flow-based NSFs that provide treatment to packets or flows such found in IPS/IDS devices, web filtering devices, flow filtering devices, deep packet inspection devices, pattern matching inspection devices, and re-mediation devices.

There are two layers of interfaces envisioned in the I2NSF approach:

o The I2NSF Capability Layer specifies how to control and monitor NSFs at a functional implementation level. This the focus for this phase of the I2NSF Work.

o The I2NSF Service Layer defines how the security policies of clients may be expressed and monitored. The Service Layer is out of scope for this phase of I2NSF’s work.

For the I2NSF capability layer, the I2NSF work proposes an interoperable protocol that passes NSF provisioning rules and orchestration information between I2NSF client on a network manager and I2NSF agent on an NSF device. It is envisioned that clients of the I2NSF interfaces include management applications, service orchestration systems, network controllers, or user applications that may solicit network security resources.

The I2NSF work to define this protocol includes the following work:

o defining an informational model that defines the concepts for standardizing the control and monitoring of NSFs,

o defining a set of Yang data models from the information model that identifies the data that must be passed,

o creating a capability registry (an IANA registry) that identifies the characteristics and behaviours of NSFs in vendor-neutral vocabulary without requiring the NSFs to be standardized.

o examining existing secure communication mechanisms to identify the appropriate ones for carrying the data that provisions and monitors information between the NSFs and their management entity (or entities).

1.2. Structure of this Document

This document provides a analysis of the gaps in the state of art in the following industry forums:

IETF working groups (section 2)

ETSI Network Functions Virtualization Industry Specification Group (ETSI NFV ISG), (section 3)

OPNFV Open Source Group (section 4)

Open Stack - Firewall as a service (OpenStack Firewall FaaS)

(section 5)
(http://docs.openstack.org/admin-guide-cloud/content/install_neutron-fwaas-agent.html)

Cloud Security Alliance Security (CSA)as a Service (section 6)

(https://cloudsecurityalliance.org/research/secaas/#_overview)

In-Depth Review of Some IETF Protocols (section 7)

1.3. Terms and Definitions

1.3.1. Requirements Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119, BCP 14 [RFC2119] and indicate requirement levels for compliant CoAP.

1.3.2. Definitions

Cloud DC: A data center that is not on premises of enterprises, but has compute/storage resources that can be requested or purchased by the enterprises. The enterprise is actually getting a virtual data center. The Cloud Security Alliance (CSA) (http://cloudsecurityalliance.org) focus on adding security to this environment. A specific research topic is security as a service within the cloud data center.

o Cloud-based security functions: Network Security Function (NSF) hosted and managed by service providers or different administrative entity.

o DC: Data Center

o Domain: The term Domain in this draft has the following different connotations in different scenarios:

* Client--Provider relationship, i.e. client requesting some

network security functions from its provider;

* Domain A - Domain B relationship, i.e. one operator domain requesting some network security functions from another operator domain; or

* Applications -- Network relationship, i.e. an application (e.g. cluster of servers) requesting some functions from network, etc.

The domain context is important because it indicates the interactions the security is focused on.

o NSF - Network Security function

o I2NSF agent - a piece of software in a device that implements a network security function which receives provisioning information and requests for operational data (monitoring data) across the I2NSF protocol from an I2NSF client.

o I2NSF client - A security client software that utilizes the I2NSF protocol to read, write or change the provisioning network security device via software interface using the I2NSF protocol (denoted as I2RS Agent)

o I2NSF Management System - I2NSF client operates within an network management system which serves as a collections and distribution point for security provisioning and filter data. This management system is denoted as I2NS management system in this document.

o Virtual Security Function: a security function that can be requested by one domain but may be owned or managed by another domain.

2. IETF Gap analysis

The IETF gap analysis first examines the IETF mechanisms which have been developed to secure the IP traffic flows through a network. Traffic filters have been defined by IETF specifications at the access points, the middle-boxes, or the routing systems. Protocols have been defined to carry provisioning and filtering traffic between a management system and an IP system (router or host system). Current security work (SACM working group (WG), MILE WG, and DOTS WG) is providing correlation of events monitored with the policy set by filters. This section provides a review the filter work, protocols, and security correlation for monitors.

2.1. Traffic Filters

2.1.1. Overview

The earliest filters defined by IETF were access filters which controlled the acceptance of IP packet data flows. Additional policy filters were created as part of the following protocols:

o COPS protocol [RFC2748] for controlling access to networks,

o Next steps in Signalling (NSIS) work (architecture: [RFC4080] protocol: [RFC5973]), and

o the Port Control Protocol (PCP) to enables IPv4 to IPv6 flexible address and port mapping for NATs and Firewalls,

Today NETMOD and I2RS Working groups are specifying additional filters in Yang modules to be used as part of the NETCONF or I2RS enhancement of NETCONF/RESTCONF.

The routing filtering is outside the scope of the flow filtering, but flow filtering may be impacted by route filtering. An initial model for the routing policy is in [I-D.shaikh-rtgwg-policy-model]

This section provides an overview of the flow filtering as an introduction to the I2NSF GAP analysis. Additional detail on NETCONF, NETMOD, I2RS, PCP, and NSIS is available in the Detailed I2NSF analysis.

2.1.1.1. Data Flow Filters in NETMOD and I2RS

The current work on expanding these filters is focused oncombining a configuration and monitoring protocol with Yang data models.

[I-D.ietf-netmod-acl-model] provides a set of access lists filters which can permit or deny traffic flow based on headers at the MAC, IP layer, and Transport layer. The configuration and monitoring protocols which can pass the filters are: NETCONF protocol [RFC6241], RESTCONF [I-D.ietf-netconf-restconf], and the I2RS protocol. The NETCONF and RESTCONF protocols install these filters into forwarding tables. The I2RS protocol uses the ACLs as part of the filters installed in an ephemeral protocol-independent filter-based RIB [I-D.kini-i2rs-fb-rib-info-model] which controls the flow of traffic on interfaces specifically controlled by the I2RS filter-based FIB.

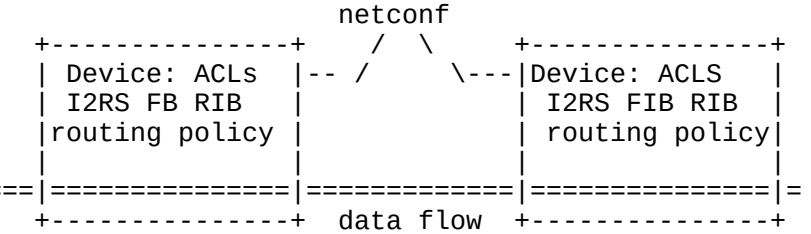
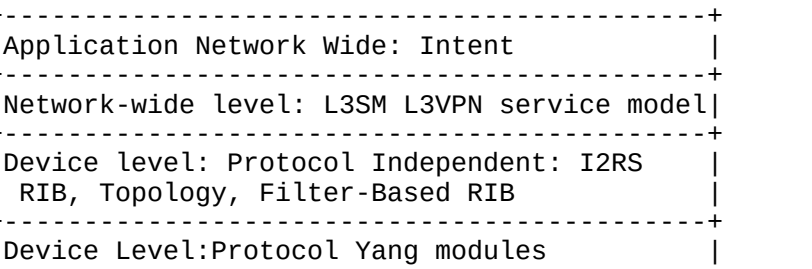


Figure 1

The I2RS protocol is a programmatic interface to the routing system. At this time, the I2RS is targeted to be extensions to the NETCONF/ RESTCONF protocols to allow the NETCONF/RESTCONF protocol to support a highly programmatic interface with high bandwidth of data, highly reliable notifications, and ephemeral state (see [I-D.ietf-i2rs-architecture]). Please see the background section on I2RS for additional details on the requirements for this extension to the NETCONF/RESTCONF protocol suite.

The vocabulary set in [I-D.ietf-netmod-acl-model] is limited, so additional protocol independent filters were written for the I2RS Filter-Based RIBs in [I-D.hares-i2rs-bnp-eca-data-model], and protocol specific filters for SFC [I-D.dunbar-i2rs-discover-traffic-rules].

One thing important to note is that NETCONF and RESTCONF manage device layer yang models. However, as figure 2 shows, there are multiple device level, network-wide level, and application level yang modules. The access lists defined by the device level forwarding table may be impacted by the routing protocols, the I2RS ephemeral protocol independent Filter-Based FIB, or some network-wide security issue (IPS/IDS).



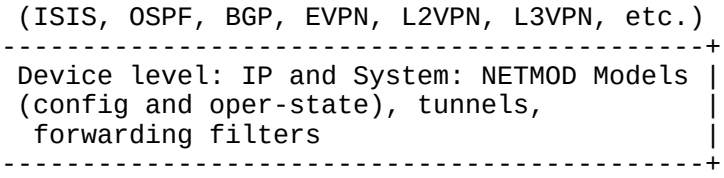


Figure 2 levels of Yang modules

2.1.1.2. I2NSF Gap analysis

The gap is that none of the current work on these filters considers all the variations of data necessary to do IPS/IDS, web-filters, stateful flow-based filtering, security-based deep packet inspection, or pattern matching with re-mediation. The I2RS Filter-Based RIB work is the closest associated work, but the focus has not been on IDS/IPS, web-filters, security-based deep packet inspection, or pattern matching with re-mediation.

The I2RS Working group (I2RS WG) is focused on the routing system so security expertise for these IDP/IPS, Web-filter, security-based deep-packet inspection has not been targeted for this WG.

Another gap is there is no capability registry (an IANA registry) that identifies the characteristics and behaviours of NSFs in vendor-neutral vocabulary without requiring the NSFs to be standardized.

What I2NSF can use from NETCONF/RESTCONF and I2RS I2NSF should consider using NETCONF/RESTCONF protocol and the I2RS proposed enhancement to the NETCONF/RESTCONF protocol.

2.1.2. Middle-box Filters

2.1.2.1. Midcom

Midcom Summary: MIDCOM developed the protocols for applications to communicate with middle boxes. However, MIDCOM have not used by the industry for a long time. This is because there was a lot of IPR encumbered technology and IPR was likely a bigger problem for IETF than it is today. MIDCOM is not specific to SIP. It was very much oriented to NAT/FW devices. SIP was just one application that needed the functionality. MIDCOM is reservation-oriented and there was an expectation that the primary deployment environment would be VoIP and real-time conferencing, including SIP, H.323, and other reservation-oriented protocols. There was an assumption that there would be some authoritative service that would have a view into endpoint sessions and be able to authorize (or not) resource allocation requests. In other word, there’s a trust model there that may not be applicable to endpoint-driven requests without some sort of trusted authorization mechanisms/tools. Therefore, there is a specific information model applied to security devices, and security device requests, that was

developed in the context of an SNMP MIB. There is also a two-stage reservation model, which was specified in order to allow better resource management.

Why I2NSF is different than Midcom

MIDCOM is different than I2NSF because its SNMP scheme doesn't work with the virtual network security functions (vNSF) management.

MidCom RFCs:

[RFC3303] - Midcom architecture

[RFC5189] - Midcom Protocol Semantics

[RFC3304] - Midcom protocol requirements

2.1.3. Security Work

2.1.3.1. Overview

Today's NSFs in security devices can handle flow-based security by providing treatment to packets/flows, such as IPS/IDS, Web filtering, flow filtering, deep packet inspection, or pattern matching and re-

mediation. These flow-based security devices are managed and provisioned by network management systems.

No standardized set of interoperable interfaces control and manage the NSFs so that a central management system can be used across security devices from multiple Vendors. I2NSF work plan is to standardize a set of interfaces by which control and management of NSFs may be invoked, operated, and monitored by:

creating an information model that defines concepts required for standardizing the control and monitoring of NSFs, and from the information model create data models. (The information model will be used to get early agreement on key technical points.)

creating a capability registry (at IANA) that enables the characteristics and behavior of NSFs to be specified using a vendor-neutral vocabulary without requiring the NSFs themselves to be standardized.

define the requirements for an I2NSF protocol to pass this traffic. (Hopefully re-using existing protocols.)

The flow-filtering configuration and management must fit into the existing security area's work plan. This section considers how the I2NSF fits into the security area work under way in the SACM (security automation and control), DOTS (DDoS Open Threat

Signalling), and MILE (Management Incident Lightweight Exchange).

2.1.3.2. Security Work and Filters

In the proposed I2NSF work plan, the I2NSF security network management system controls many NSF nodes via the I2NSF Agent. This control of data flows is similar to the COPS example in section x.x.

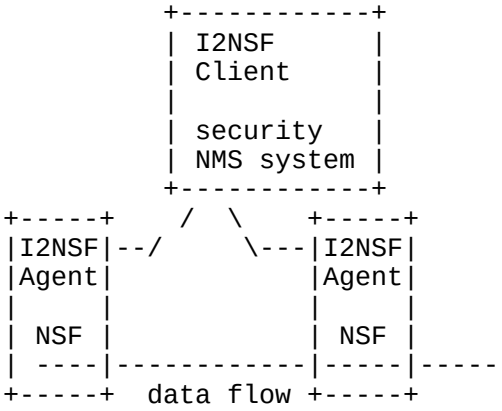


Figure 2

The other security protocols work to interact within the network to provide additional information in the following way:

o SACM [I-D.ietf-sacm-architecture] describes an architecture which tries to determine if the end-point security policies and the reality (denoted as security posture) align. [I-D.ietf-sacm-terminology] defines posture as the configuration and/or status of hardware or software on an endpoint as it pertains to an organization's security policy. Filters can be considered on the configuration or status pieces that needs to be monitored.

o DOTS (DDoS Open Threat Signalling) - is working on coordinating the mitigation of DDoS attacks. A part of DDoS attach mitigation is to provide lists of addresses to be filtered via IP header filters.

o MILE (Managed Incident LIghtweight Exchange) - is working on creating a standardized format for incident and indicator reports, and creating a protocol to transport this information. The incident information MILE collects may cause changes in data-flow filters on one or more NSFs.

2.1.3.3. I2NSF interaction

The network management system that the I2NSF client resides on may interact with other clients or agents developed for the work ongoing in the SACM, DOTS, and MILES working groups. This section describes how the addition of I2NSF's ability to control and monitor NSF

devices is compatible and synergistic with these existing efforts.

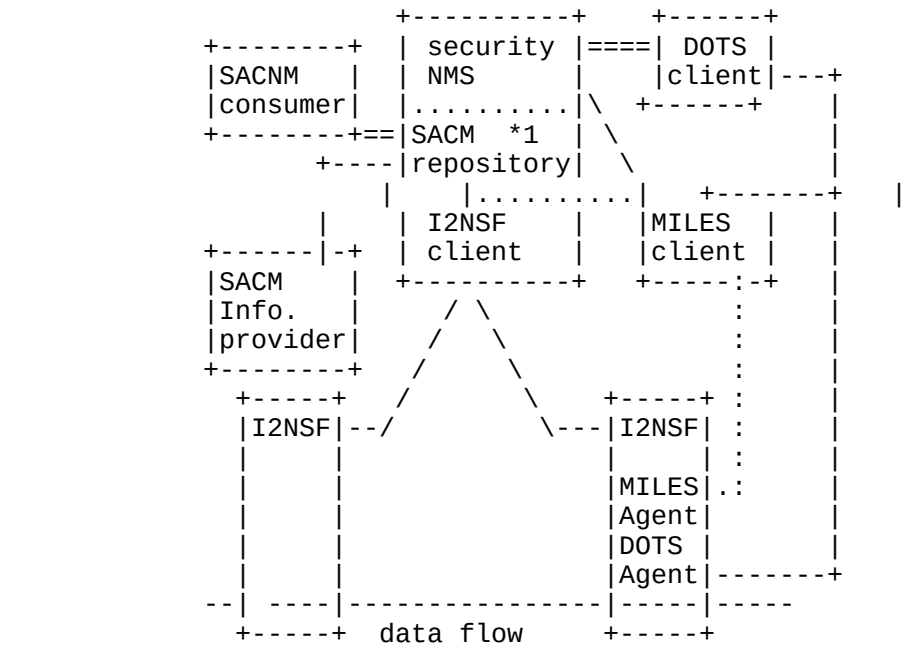


Figure 3

Figure 3 provides a diagram of a system the I2NSF, SACM, DOTS and MILES client-agent or consumer-broker-provider are deployed together. The following are possible positive interactions these scenario might have:

o An security network management system (NMS) can contain a SACM repository and be connected to SACM information provider and a SACM consumer. The I2NSF may provide one of the ways to change the forwarding filters.

o The security NMS may also be connected to DOTS DDoS clients managing the information and configuring the rules. The I2NSF may provide one of the ways to change forwarding filters.

o The MILES client on a security network management system talking to the MILES agent on the node may react to the incidents by using I2NSF to set filters. DOTS creates black-lists, but does not have a complete set of filters.

2.1.3.4. Benefits from the Interaction

I2NSF's ability to provide a common interoperable and vendor neutral interface may allow the security NMS to use a single change to change filters. SACM provides an information model to describe end-points, but does not link this directly to filters.

DOTS creates black-lists based on source and destination IP address, transport port number, protocol ID, and traffic rate. Like NETMOD's, ACLS are not sufficient for all filters or control desired by the NSF boxes.

The incident data captured by MILES will not have enough filter information to provide NSF devices with general services. The I2NSF will be able to handle the MILE incident data and create alerts or reports for other security systems.

3. ETSI NFV

3.1. ETSI Overview

Network Function Virtualization (NFV) provides the service providers with flexibility, cost effective and agility to offer their services to customers. One such service is the network security function which guards the exterior of a service provider or its customers.

The flexibility and agility of NFV encourages service providers to provide different products to address business trends in their market to provide better service offerings to their end user. A traditional product such as the network security function (NSF) may be broken into multiple virtual devices each hosted from another vendor. In the past, network security devices may have been single sourced from a small set of vendors - but in the NFV version of NSF devices, this reduced set of sources will not provide a competitive edge. Due to this market shift, the network security device vendors are realizing that the proprietary provisioning protocols and formats of data may be a liability. Out of the NFV work has arisen a desire for a single interoperable network security device provisioning and control protocol.

The I2NSF will be deployed along networks using other security and NFV technology. As section 3 described, the NFV NSF security is deployed along side other security functions (AAA, SACM, DOTS, and MILE devices) or deep-packet-inspection. The ETSI Network Functions Virtualization: NFV security: Security and Trust guidance document (ETSI NFV SEC 003 1.1.1 (2014-12)) indicates that multiple administrative domains will deployed in carrier networks. One example of these multiple domains is hosting of multiple tenant domains (telecom service providers) on a single infrastructure domain (infrastructure service) as figure 4 shows. The ETSI Inter inter-VNFM document (aka Ve-Vnfn) between the element management system and the Virtual network function is the equivalent of the interface between the I2NSF client on a management system and the I2NSF agent

on the network security feature VNF.

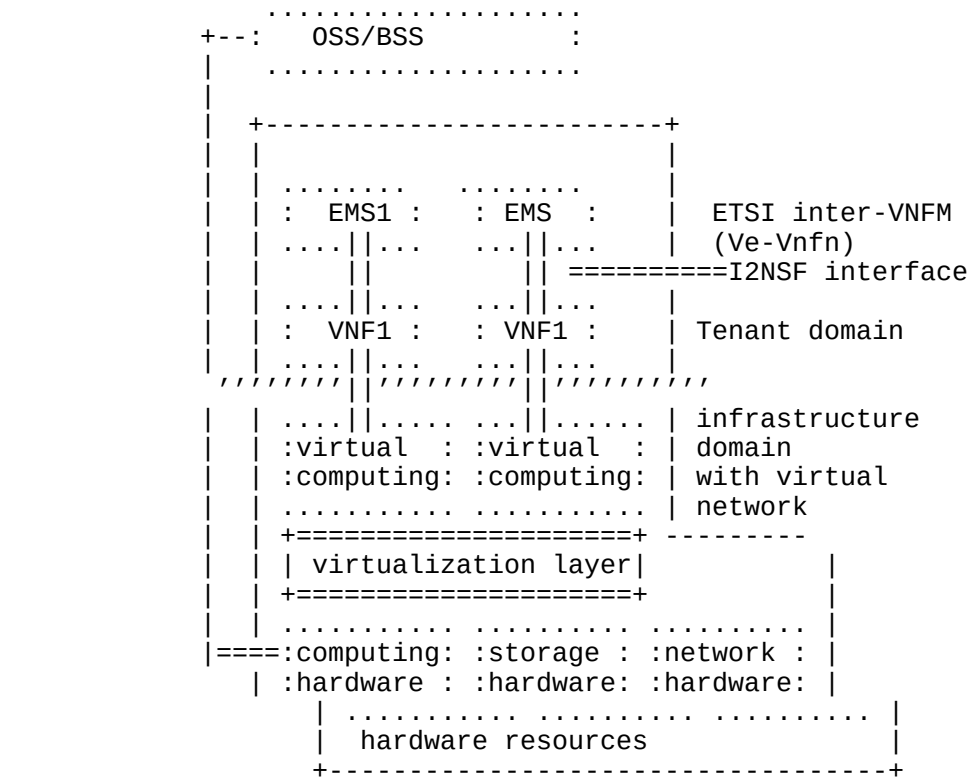


figure 4

The ETSI proof of concept work has worked on the following security proof of concepts:

o #16 - NFVIaas with Secure, SDN controlled WAN Gateway,

3.2. I2NSF Gap Analysis

The I2NSF will be deployed on top of virtual computing linked together by virtual routers configured by NETCONF/RESTCONF or I2RS which provision and monitoring the L1, L2, l3 and service pathways through the network.

In the NFV-related productions, the current architecture does not have a protocol to maintain an interoperability provisioning from I2NSF client to I2NSF agent. The result is that service providers have to manage the interoperability using private protocols. In response to this problem, the device manufacturers and the service providers have begun to discuss an I2NSF protocol for interoperable passing of provisioning and filter in formation.

Open source work (such as OPNFV) provides a common code base for

providers to start their NFV work from. However, this code base faces the same problem. There is no defacto standard protocol.

4. OPNFV

The OPNFV (www.opnfv.org) is a carrier-grade integrated, open source platform focused on accelerating the introduction of new Network Function Virtualization (NFV) products and service. The OPNFV Moon project is focused on adding the security interface for a network management system within the Tenant NFVs and the infrastructure NFVs (as shown in figure 4). This section provides an overview of the OPNFV Moon project and a gap analysis between I2NSF and the OPNFV Moon Project.

4.1. OPNFV Moon Project

The OPNFV moon project (https://wiki.opnfv.org) is a security management system. NFV uses cloud computing technologies to virtualize the resources and automate the control. The Moon project is working on a security manager for the Cloud computing infrastructure (https://wiki.opnfv.org/moon). The Moon project proposes to provision a set of different cloud resources/services for VNFs (Virtualized Network Functions) while managing the isolation of VNS, protection of VNFs, and monitoring of VNS. Moon is creating a security management system for OPNFV with security managers to protect different layers of the NFV infrastructure. The Moon project is choosing various security project mechanisms "a la cart" to enforcement related security managers. A security management system integrates mechanisms of different security aspects. This project will first propose a security manager that specifies users' security requirements. It will also enforce the security managers through various mechanisms like authorization for access control, firewall for networking, isolation for storage, logging for tractability, etc.

The Moon security manager operates a VNF security manager at the ETSI VeVnfm level where the I2NSF protocol is targeted as figure 5 shows. This figure also shows how the OPNFV VNF Security project mixes the I2NSF level with the device level.

The Moon project lists the following gaps in OpenStack:

o No centralized control for compute, storage, and networking. Open Stack uses Nova for computing and Swift for software. Each system has a configuration file and its own security policy. This lacks the synchronization mechanism to build a complete secure configuration for OPNF.

o No dynamic control so that if a user obtains the token, the is no way to obtain control over the user.

o No customization or flexibility to allow integration into different vendors,

o No fine grain authorization at user level. Authorization is only at the API

Moon addresses these issues adding authorization, logging, IDS, enforcement of network policy, and storage protection. Moon is based on OpenStack Keystone.

Deliverable time frame: 2S 2015

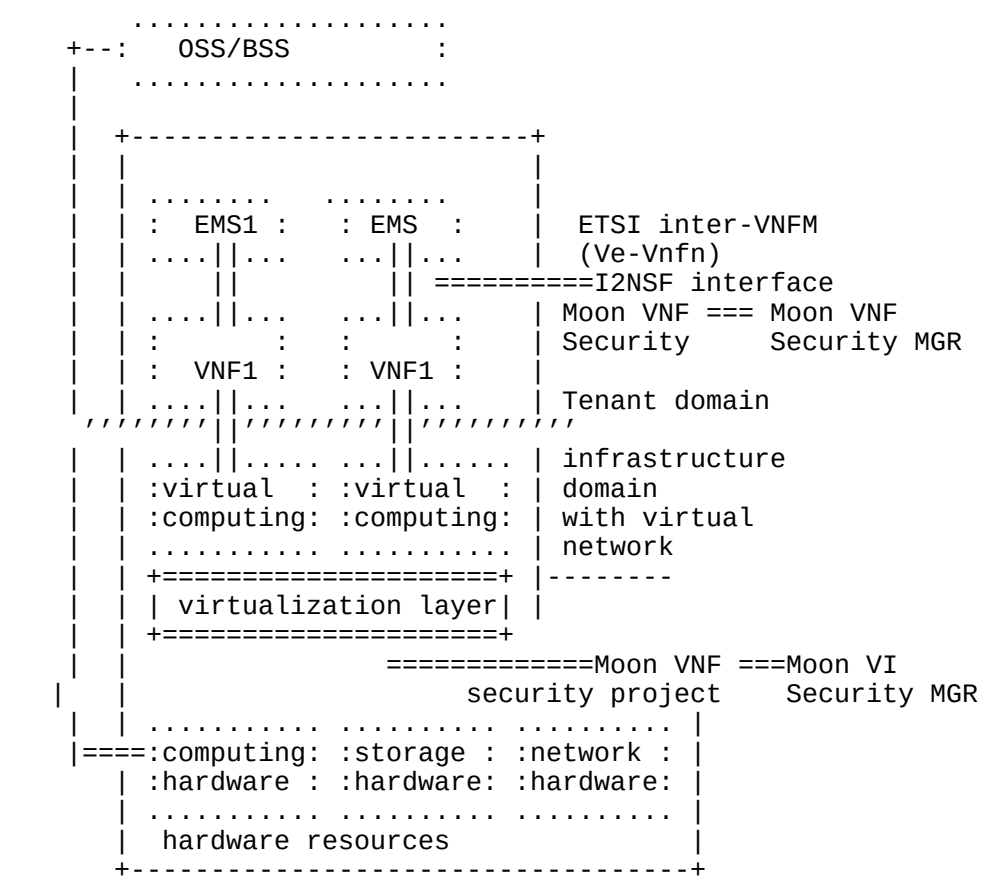


figure 5

4.2. Gap Analysis for OPNFV Moon Project

OpenStack congress does not provide vendor independent systems.

5. OpenStack Security Firewall

OpenStack has advanced features of: a) API for managing security groups (http://docs.openstack.org/admin-guide-cloud/content/section_securitygroups.html) and b) firewalls as a service (http://docs.openstack.org/admin-guide-cloud/content/fwaas_api_abstractions.html).

This section provides an overview of this open stack work, and a gap analysis of how I2NSF provides additional functions

5.1. Overview of API for Security Group

The security group with the security group rules provides ingress and egress traffic filters based on port. The default group drops all ingress traffic and allows all egress traffic. The groups with additional filters are added to change this behaviour. To utilize the security groups, the networking plug-in for Open Stack must implement the security group API. The following plug-ins in OpenSTsack currently implement this security: ML2, Open vSwitch, Linux Bridge, NEC, and VMware NSX. In addition, the correct firewall driver must be added to make this functional.

5.2. Overview of Firewalls as a Service

Firewall as a service is an early release of an API that allows early adopters to test network implementations. It contains APIs with parameters for firewall rules, firewall policies, and firewall identifiers. The firewall rules include the following information:

- identification of rule (id, name, description)
- identification tenant rule associated with,
- links to installed firewall policy,
- IP protocol (tcp, udp, icmp, none)
- source and destination IP address
- source and destination port
- action: allow or deny traffic
- status: position and enable/disabled

The firewall policies include the following information:

- identification of the policy (id, name, description),
- identification of tenant associated with,

o ordered list of firewall rules,

o indication if policy can be seen by tenants other than owner, and

o indication if firewall rules have been audited.

The firewall table provides the following information:

o identification of firewall (id, name, description),

o tenant associated with this firewall,

o administrative state (up/down),

o status (active, down, pending create, pending delete, pending update, pending error)

o firewall policy ID this firewall is associated with

5.3. I2NSF Gap analysis

The OpenStack work is preliminary (security groups and firewall as a service). This work does not allow any of the existing network security vendors provide a management interface. Security devices take time to be tested for functionality and their detection of security issues. The OpenStack work provides an interesting simple set of filters, and may in the future provide some virtual filter service. However, at this time this open source work does not address the single management interfaces for a variety of security devices.

I2NSF is proposing rules that will include Event-Condition-matches (ECA) with the following matches packet based matches on L2, L3, and L4 headers and/or specific addresses within these headers, context based matches on schedule state and schedule, [Editor: Need more details here.]

The I2NSF is proposing action for these ECA policies of:

basic actions of deny, permit, and mirror,

advanced actions of: IPS signature filtering and URL filtering.

6. CSA Secure Cloud

6.1. CSA Overview

The Cloud Security Alliance (CSA)(www.cloudsecurityaliance.org) defined security as a service (SaaS) in their Security as a Service working group (SaaS WG) during 2010-2012. The CSA SaaS group defined

ten categories of network security
(https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_V1_0.pdf)
and provides implementation guidance for each of these ten categories
This section provides an overview of the CSA SaaS working groups
documentation and a Gap analysis for I2NSF

6.1.1. CSA Security as a Service(SaaS)

The CSA SaaS working group defined the following ten categories, and provided implementation guidance on these categories:

1. Identity Access Management (IAM)
(https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_1_IAM_Implementation_Guidance.pdf)

2. Data Loss Prevention (DLP)
(https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_2_DLP_Implementation_Guidance.pdf)

3. Web Security (web)
(https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_3_Web_Security_Implementation_Guidance.pdf),

4. Email Security (email)
(https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_4_Email_Security_Implementation_Guidance.pdf),

5. Security Assessments
(https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_5_Security_Assessments_Implementation_Guidance.pdf),

6. Intrusion Management
(https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_6_Intrusion_Management_Implementation_Guidance.pdf),

7. Security information and Event Management
(https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_7_SIEM_Implementation_Guidance.pdf),

8. Encryption
(https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_8_Encryption_Implementation_Guidance.pdf),

9. Business Continuity and Disaster Recovery (BCDR)
https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_9_BCDR_Implementation_Guidance.pdf), and

10. Network Security
(https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_10_Network_Security_Implementation_Guidance.pdf).

The sections below give an overview these implementation guidances

6.1.2. Identity Access Management (IAM)

document:

(https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_1_IAM_Implementation_Guidance.pdf)

The identity management systems include the following services:

- Centralized Directory Services,
- Access Management Services,
- Identity Management Services,
- Identity Federation Services,
- Role-Based Access Control Services,
- User Access Certification Services,
- Privileged User and Access Management,
- Separation of Duties Services, and
- Identity and Access Reporting Services.

The IAM device communications with the security management system that controls the filtering of data. The CSA SaaS IAM specification states that interoperability between IAM devices and secure access network management systems is a a problem. This 2012 implementation report confirms there is a gap with I2NSF

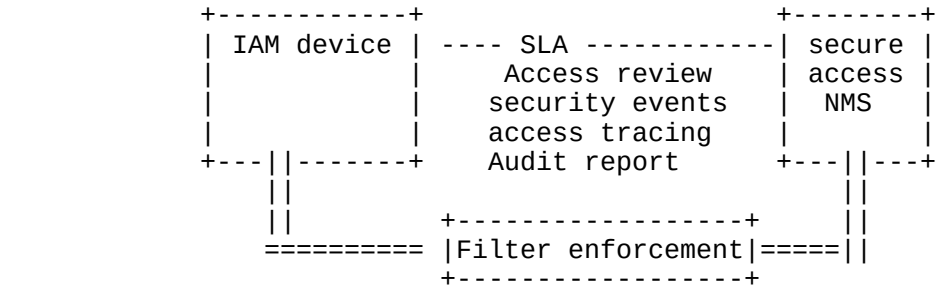


Figure 6

6.1.3. Data Loss Prevention (DLP)

Document:

(https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_2_DLP_Implementation_Guidance.pdf)

The data loss prevention (DLP)services must address:

o origination verification,

o integrity of data,

o confidentiality and access control,

o accountability,

o avoiding false positives on detection, and

o privacy concerns.

The CSA SaaS DLP device communications require that it have the enforcement capabilities to do the following:

alert and log data loss,

delete data on system or passing through,

filter out (block/quarantine) data,

reroute data,

encrypt data

Figure 7

6.1.4. Web security(Web))

Document:

https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_3_Web_Security_Implementation_Guidance.pdf

The web security services must address:

o Web 2.0/Social Media controls,

o Malware and Anti-Virus controls,

o Data Loss Prevention controls (over Web-based services like Gmail or Box.net),

o XSS, JavaScript and other web specific attack controls

o Web URL Filtering,

o Policy control and administrative management,

o Bandwidth management and quality of service (QoS) capability, and

o Monitoring of SSL enabled traffic.

The CSA SaaS Web services device communications require that it have the enforcement capabilities to do the following:

alert and log malware or anti-virus data patterns,

delete data (malware and virus) passing through systems,

filter out (block/quarantine) data,

filter Web URLs,

interact with policy and network management systems,

control bandwidth and QoS of traffic, and

monitor encrypted (SSL enabled) traffic,

All of these features either require the I2NSF standardized I2NSF client to I2NSF agent to provide multi-vendor interoperability.

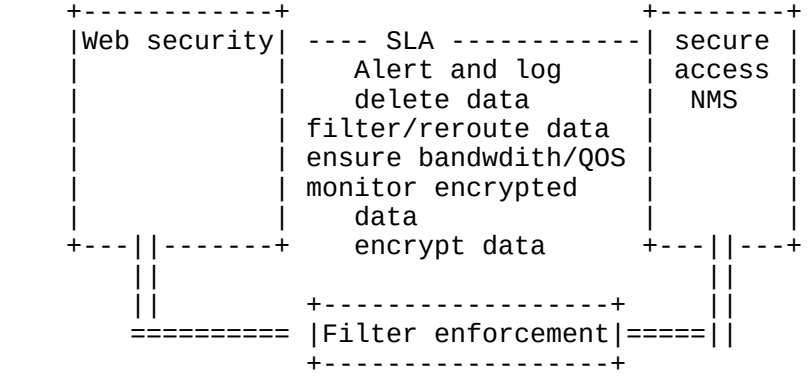


Figure 8

6.1.5. Email Security (email))

Document:

https://downloads.cloudsecurityalliance.org/initiatives/secaas/

SecaaS_Cat_4_Email_Security_Implementation_Guidance.pdf

The CSA Document recommends that email security services must address:

o Common electronic mail components,

o Electronic mail architecture protection,

o Common electronic mail threats,

o Peer authentication,

o Electronic mail message standards,

o Electronic mail encryption and digital signature,

o Electronic mail content inspection and filtering,

o Securing mail clients, and

o Electronic mail data protection and availability assurance techniques

The CSA SaaS Email security services requires that it have the enforcement capabilities to do the following:

provide the malware and spam detection and removal,

alert and provide rapid response to email threats,

identify email users and secure remote access to email,

do on-demand provisioning of email services,

filter out (block/quarantine) email data,

know where the email traffic or data is residing (to to regulatory issues), and

be able to monitor encrypted email,

be able to encrypt email,

be able to retain email records (while abiding with privacy concerns), and

interact with policy and network management systems.

All of these features require the I2NSF standardized I2NSF client to I2NSF agent to provide multi-vendor interoperability.

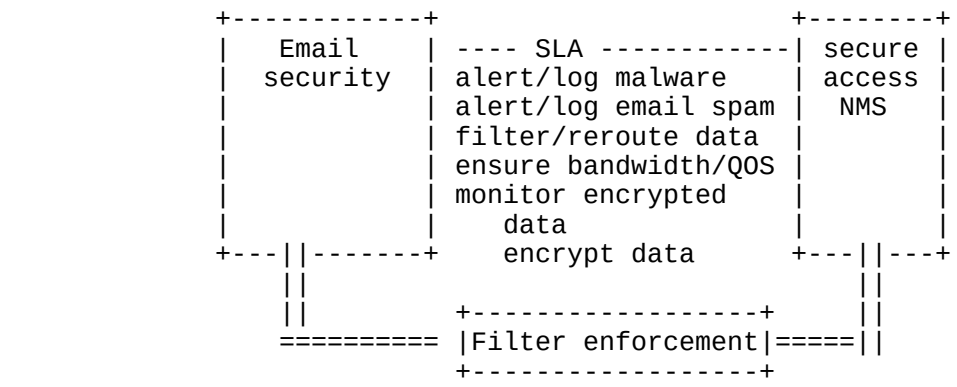


Figure 9

6.1.6. Security Assessment

Document:

https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_5_Security_Assessments_Implementation_Guidance.pdf

The CSA SaaS Security assessment indicates that assessments need to be done on the following devices:

o hypervisor infrastructure,

o network security compliance systems,

o Servers and workstations,

o applications,

o network vulnerabilities systems,

o internal auditor and intrusion detection/prevention systems (IDS/IPS), and

o web application systems.

All of these features require the I2NSF working group standardize the way to pass these assessments to and from the I2NSF client on the I2NSF management system and the I2NSF Agent.

6.1.7. Intrusion Detection

Document:

https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_6_Intrusion_Management_Implementation_Guidance.pdf)

The CSA SaaS Intrusion detection management includes intrusion detection through: devices:

o Network traffic inspection, behavioural analysis, and flow analysis,

o Operating System, Virtualization Layer, and Host Process Events monitoring,

o monitoring of Application Layer Events, and

o Correlation Techniques, and other Distributed and Cloud-Based Capabilities

Intrusion response includes both:

o Automatic, Manual, or Hybrid Mechanisms,

o Technical, Operational, and Process Mechanisms.

The CSA SaaS recommends the intrusion security management systems include provisioning and monitoring of all of these types of intrusion detection (IDS) or intrusion protection devices. The management of these systems requires also requires:

Central reporting of events and alerts,

administrator notification of intrusions,

Mapping of alerts to Cloud-Layer Tenancy,

Cloud sourcing information to prevent false positives in detection, and

allowing for redirection of traffic to allow remote storage or transmission to prevent local evasion.

All of these features require the I2NSF standardized I2NSF client to I2NSF agent to provide multi-vendor interoperability.

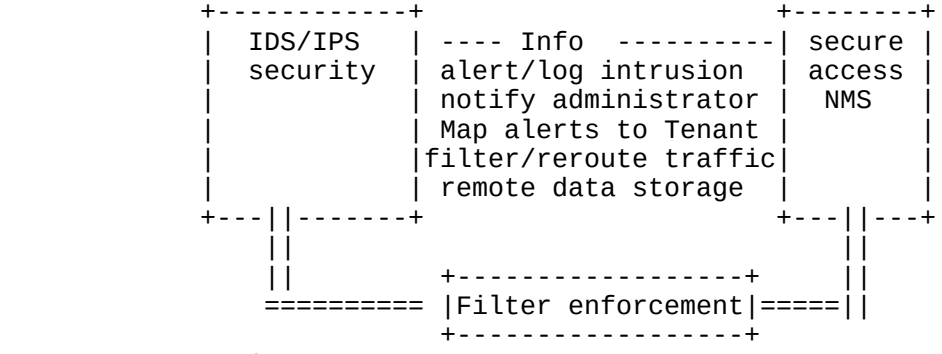


Figure 10

6.1.8. Security Information and Event Management(SEIM)

Document:

https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_7_SIEM_Implementation_Guidance.pdf)

The Security Information and Event Management (SEIM) receives data from a wide range of security systems such as Identity management systems (IAM), data loss prevention (DLP), web security (Web), email security (email), intrusion detection/prevision (IDS/IPS)), encryption, disaster recovery, and network security. The SEIM combines this data into a single streams. All the requirements for data to/from these systems are replicated in these systems needs to give a report to the SIEM system.

A SIEM system would be prime candidate to have a I2NSF client that gathers data from an I2NSF Agent associated with these various types of security systems. The CSA SaaS SIEM functionality document

suggests that one concern is to have standards that allow timely recording and sharing of data. I2NSF can provide this.

6.1.9. Encryption

Document:

https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_8_Encryption_Implementation_Guidance.pdf

The CSA SaaS Encryption implementation guidance document considers how one implements and manages the following security systems:

key management systems (KMS), control of keys, and key life cycle;

Shared Secret encryption (Symmetric ciphers),

No-Secret or Public Key Encryption (asymmetric ciphers),

hashing algorithms,

Digital Signature Algorithms,

Key Establishment Schemes,

Protection of Cryptographic Key Material (FIPS 140-2; 140-3),

Interoperability of Encryption Systems, Key Conferencing, Key Escrow Systems, and others

application of Encryption for Data at rest, data in transit, and data in use;

PKI (including certificate revocation "CRL");

Future application of such technologies as Homomorphic encryption, Quantum Cryptography, Identitybased Encryption, and others;

Crypto-system Integrity (How bad implementations can under mind a crypto-system), and

Cryptographic Security Standards and Guidelines

The wide variety of encryption services require the security management systems be able to provision, monitor, and control the systems that are being used to encrypt data. This document indicates in the implementation sections that the standardization of interfaces to/from management systems are key to good key management systems, encryption systems, and crypto-systems.

6.1.10. Business Continuity and Disaster Recovery (BC/DR)

Document:

https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_9_BCDR_Implementation_Guidance.pdf

The CSA SaaS Business Continuity and Disaster Recovery (BC/DR)

implementation guidance document considers the systems that implement the the contingency plans and measures designed and implemented to ensure operational resiliency in the event of any service interruptions. BC/DR systems includes:

Business Continuity and Disaster Recovery BC/DR as a service, including categories such as complete Disaster Recovery as a Service (DRaaS), and subsets such as file recovery, backup and archive,

Storage as a Service including object, volume, or block storage;

old Site, Warm Site, Hot Site backup plans;

IaaS (Infrastructure as a Service), PaaS (Platform as a Service), and SaaS (Software as a Service);

Insurance (and insurance reporting programs)

Business Partner Agents (business associate agreements);

System Replication (for high availability);

Fail-back to Live Systems mechanisms and management;

Recovery Time Objective (RTO) and Recovery Point Objective (RPO);

Encryption (data at rest [DAR], data in motion [DIM], field level);

Realm-based Access Control;

Service-level Agreements (SLA); and ISO/IEC 24762:2008, BS25999, ISO 27031, and FINRA Rule 4370

These BC/DR systems must handle data backup and recovery, server backup/recovery, and data center (virtual/physical) backup and recovery. Recovery as a service (RaaS) means that the BC/DR services are being handled by management systems outside the enterprise.

The wide variety of BC/DR requires the security management systems to be able to communicate provisioning, monitor, and control those systems that are being used to back-up and restore data. An interoperable protocol that allows provision and control of data center's data, servers, and data center management devices devices is extremely important to this application. Recovery as a Service (SaaS) indicates that these services need to be able to be remotely management.

The CSA SaaS BC/BR documents indicate how important a standardized I2NSF protocol is.

6.1.11. Network Security Devices

Document:

https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_10_Network_Security_Implementation_Guidance.pdf

The CSA SaaS Network Security implementation recommendation includes advice on:

How to segment networks,

Network security controls,

Controlling ingress and egress controls such as Firewalls (Stateful), Content Inspection and Control (Network-based), Intrusion Detection System/Intrusion Prevention Systems (IDS/IPS), and Web Application Firewalls,

Secure routing and time,

Denial of Service (DoS) and Distributed Denial of Service (DDoS) Protection/Mitigation,

Virtual Private Network (VPN) with Multiprotocol Label Switching (MPLS) Connectivity (over SSL), Internet Protocol Security (IPsec) VPNs, Virtual Private LAN Service (VPLS), and Ethernet Virtual Private Line (EVPL),

Threat Management,

Forensic Support, and

Privileged User/Use Monitoring.

These network security systems require provisioning, monitoring, and the ability for the security management system to subscribe to

receive logs, snapshots of capture data, and time synchronization. This document states the following:

"It is critical to understand what monitoring APIs are available from the CSP, and if they match risk and compliance requirements",

"Network security auditors are challenged by the need to track a server and its identity from creation to deletion. Audit tracking is challenging in even the most mature cloud environments, but the challenges are greatly complicated by cloud server sprawl, the situation where the number of cloud servers being created is growing more quickly than a cloud environments ability to manage them."

A valid threat vector for cloud is the API access. Since a majority of CSPs today support public API interfaces available within their networks and likely over the Internet."

The CSA SaaS network security indicates that the I2NSF must be secure so that the I2NSF Client-Agent protocol does not become a valid threat vector. In additions, the need for the management protocol like I2NSF is critical in the sprawl of Cloud environment.

6.2. I2NSF Gap Analysis

The CSA Security as a Service (SaaS) document show clearly that there is a gap between the ability of the CSA SaaS devices to have a vendor neutral, inoperable protocol that allow the multiple of network security devices to communicate passing provisioning and informational data. Each of the 10 implementation agreements points to this as a shortage. The I2NSF yang models and protocol is needed according to the CSA SaaS documents.

7. In-depth Review of IETF protocols

7.1. NETCONF and RESTCONF

The IETF NETCONF working group has developed the basics of the NETCONF protocol focusing on secure configuration and querying operational state. The NETCONF protocol [RFC6241] may be run over TLS [RFC6639] or SSH ([RFC6242]. NETCONF can be expanded to defaults

[RFC6243], handling events ([RFC5277] and basic notification [RFC6470], and filtering writes/reads based on network access control models (NACM, [RFC6536]). The NETCONF configuration must be committed to a configuration data store (denoted as config=TRUE). Yang models identify nodes within a configuration data store or an operational data store using a XPath expression (document root ---to --- target source). NETCONF uses an RPC model and provides protocol for handling configs (get-config, edit-config, copy-config, delete- config, lock, unlock, get) and sessions (close-session, kill- session). The NETCONF Working Group has developed RESTCONF, which is an HTTP-based protocol that provides a programmatic interface for accessing data defined in YANG, using the datastores defined in NETCONF.

RESTCONF supports "two edit condition detections" - time stamp and entity tag. RESTCONF uses a URI encoded path expressions. RESTCONF provides operations to get remote servers options (OPTIONS), retrieve data headers (HEAD), get data (GET), create resource/invoke operation (POST), patch data (PATCH), delete resource (DELETE), or query. RFCs for NETCONF

- o NETCONF [RFC6242]
- o NETCONF monitoring [RFC6022]
- o NETCONF over SSH [RFC6242]
- o NETCONF over TLS [RFC5539]
- o NETCONF system notification> [RFC6470]
- o NETCONF access-control (NACM) [RFC6536]
- o RESTCONF [I-D.ietf-netconf-restconf]
- o NETCONF-RESTCONF call home [I-D.ietf-netconf-call-home]
- o RESTCONF collection protocol [I-D.ietf-netconf-restconf-collection]
- o NETCONF Zero Touch Provisioning [I-D.ietf-netconf-zerotouch]

7.2. I2RS Protocol

Based on input from the NETCONF working group, the I2RS working group decided to re-use the NETCONF or RESTCONF protocols and specify additions to these protocols rather than create yet another protocol (YAP).

The required extensions for the I2RS protocol are in the following drafts:

- o Ephemeral state [I-D.ietf-i2rs-ephemeral-state],

o Publication-Subscription notifications [I-D.ietf-i2rs-pub-sub-requirements],

o Traceability [I-D.ietf-i2rs-traceability],

o Security requirements [I-D.hares-i2rs-auth-trans]

At this time, NETCONF and RESTCONF cannot handle the ephemeral data store proposed by I2RS, the publication and subscription requirements, the traceability, or the security requirements for the transport protocol and message integrity.

7.3. NETMOD Yang modules

NETMOD developed initial Yang models for interfaces [RFC7223]), IP address ([RFC7277]), IPv6 Router advertisement ([RFC7277]), IP Systems ([RFC7317]) with system ID, system time management, DNS resolver, Radius client, SSH, syslog ([I-D.ietf-netmod-syslog-model]), ACLS ([I-D.ietf-netmod-acl-model]), and core routing blocks ([I-D.ietf-netmod-routing-cfg] The routing working group (rtgwg) has begun to examine policy for routing and tunnels.

Protocol specific Working groups have developed yang models for ISIS ([I-D.ietf-isis-yang-isis-cfg]), OSPF ([I-D.ietf-ospf-yang]), and BGP (merge of [I-D.shaikh-idr-bgp-model] and [I-D.zhdankin-idr-bgp-cfg] with the bgp policy proposed multiple Working groups (idr and rtgwg)). BGP Services yang models have been proposed for PPB EVPN ([I-D.tsingh-bess-pbb-evpn-yang-cfg]), EVPN ([I-D.zhuang-bess-evpn-yang]), L3VPN ([I-D.zhuang-bess-l3vpn-yang]), and multicast MPLS/BGP IP VPNs ([I-D.liu-bess-mvpn-yang]).

7.4. COPS

One early focus on flow filtering based on policy enforcement of traffic entering a network is the 1990s COPS [RFC2748] design (PEP and PDP) as shown in figure 1. The Policy decision point kept network-wide policy (E.g. ACLs) and sent it to Policy enforcements who then would control what data flows between the two These decision points controlled data flow from PEP to PEP. [RFC3084] describes COPS use for policy provisioning.

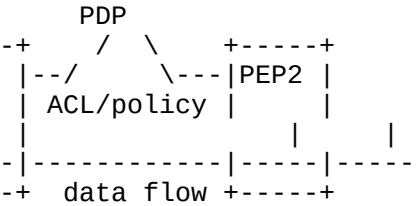


Figure 11

COPS had a design of Policy Enforcement Points (PEP), and policy Decision Points (PDP) as shown in figure 11. These decision points controlled flow from PEP to PEP.

Why COPS is no longer used

Security in the network in 2015 uses specific devices (IDS/IPS, NAT firewall, etc) with specific policies and profiles for each types of device. No common protocol or policy format exists between the policy manager (PDP) and security enforcement points.

COPs RFCs: [RFC4261], [RFC2940], , [RFC3084], , [RFC3483]

Why I2NSF is different COPS

COPS was a protocol for policy related to Quality of Service (QoS) and signalling protocols (e.g. RSVP) (security, flow, and others). I2NSF creates a common protocol between security policy decision points (SPDP) and security enforcement points (SEP). Today's security devices currently only use proprietary protocols. Manufacturers would like a security specific policy enforcement protocol rather than a generic policy protocol.

7.5. PCP

As indicated by the name, the Port Control Protocol (PCP) enables an IPv4 or IPv6 host to flexibly manage the IP address and port mapping information on Network Address Translators (NATs) or firewalls, to facilitate communication with remote hosts.

PCP RFCs:

[RFC6887]

[RFC7225]

[I-D.ietf-pcp-authentication]

[I-D.ietf-pcp-optimize-keepalives]

[I-D.ietf-pcp-proxy]

Why is I2NSF different from PCP:

Here are some aspects that I2NSF is different from PCP:

o PCP only supports the management of port and address information rather than any other security functions

o Cover the proxy, firewall and NAT box proposals in I2NSF

7.6. NSIS - Next steps in Signalling

NSIS is for standardizing an IP signalling protocol (RSVP) along data path for end points to request its unique QoS characteristics, unique FW policies or NAT needs (RFC5973) that are different from the FW/NAT original setting. The requests are communicated directly to the FW/ NAT devices. NSIS is like east-west protocols that require all involved devices to fully comply to make it work.

NSIS is path-coupled, it is possible to message every participating device along a path without having to know its location, or its location relative to other devices (this is particularly a pressing issue when you've got one or more NATs present in the network, or when trying to locate appropriate tunnel endpoints).

A diagram should be added here showing I2NSF and NSIS

Why I2NSF is different than NSIS:

o The I2NSF requests from clients do not go directly to network security devices, but instead to controller or orchestrator that can translate the application/user oriented policies to the involved devices in the interface that they support.

o The I2NSF request does not require all network functions in a path to comply, but it is a protocol between the I2NSF client and the I2NSF Agent in the controller and orchestrator

o I2NSF defines client (applications) oriented descriptors (profiles, or attributes) to request/negotiate/validate the network security functions that are not on the local premises.

Why we belief I2NSF has a higher chance to be deployed than NSIS:

o Open Stack already has a proof-of-concept/preliminary implementation, but the specification is not complete. IETF can play an active role to make the specification for I2NSF is complete. IETF can complete and extend the OpenStack implementation to provide an interoperable specification that can meet the needs and requirements of operators and is workable for suppliers of the technology. The combination of a carefully designed interoperable IETF specification with an open-source code development Open Stack will leverage the strengths of the two communities, and expand the informal ties between the two groups. A software development cycle has the following components: architecture, design specification, coding, and interoperability testing. The IETF can take ownership of the first two steps, and provide expertise and a good working atmosphere (in hack-a-thons) in the last two steps for OpenSTack or other open-source coders.

o IETF has the expertise in security architecture and design for interoperable protocols that span controllers/routers, middle- boxes, and security end-systems.

o IETF has a history of working on interoperable protocols or virtualized network functions (L2VPN, L3VPN) that are deployed by operators in large scale devices. IETF has a strong momentum to create virtualized network functions (see SFC WG in routing) to be deployed in network boxes. [Note: We need to add SACM and others here].

8. Security Considerations

There is no security consideration

9. IANA Considerations

There is no IANA consideration

10. References

10.1. Normative References

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

10.2. Informative References

[I-D.dunbar-i2rs-discover-traffic-rules] Dunbar, L. and S. Hares, "An Information Model for Filter Rules for Discovery and Traffic for I2RS Filter-Based RIB", draft-dunbar-i2rs-discover-traffic-rules-00 (work in progress), March 2015.

[I-D.hares-i2rs-auth-trans] Hares, S., Migault, D., and J. Halpern, "I2RS Security Related Requirements", draft-hares-i2rs-auth-trans-04 (work in progress), July 2015.

[I-D.hares-i2rs-bnp-eca-data-model] Hares, S., Wu, Q., Tantsura, J.,

and R. White, "An Information Model for Basic Network Policy and Filter Rules", draft-hares-i2rs-bnp-eca-data-model-00 (work in progress), July 2015.

[I-D.hares-i2rs-info-model-service-topo] Hares, S., Wu, W., Wang, Z., and J. You, "An Information model for service topology", draft-hares-i2rs-info-model-service-topo-03 (work in progress), January 2015.

[I-D.ietf-i2rs-architecture] Atlas, A., Halpern, J., Hares, S., Ward, D., and T. Nadeau, "An Architecture for the Interface to the Routing System", draft-ietf-i2rs-architecture-09 (work in progress), March 2015.

[I-D.ietf-i2rs-ephemeral-state] Haas, J. and S. Hares, "I2RS Ephemeral State Requirements", draft-ietf-i2rs-ephemeral-state-00 (work in progress), June 2015.

[I-D.ietf-i2rs-problem-statement] Atlas, A., Nadeau, T., and D. Ward, "Interface to the Routing System Problem Statement", draft-ietf-i2rs-problem-statement-06 (work in progress), January 2015.

[I-D.ietf-i2rs-pub-sub-requirements] Voit, E., Clemm, A., and A. Prieto, "Requirements for Subscription to YANG Datastores", draft-ietf-i2rs-pub-sub-requirements-02 (work in progress), March 2015.

[I-D.ietf-i2rs-rib-data-model] Wang, L., Ananthakrishnan, H., Chen, M., amit.dass@ericsson.com, a., Kini, S., and N. Bahadur, "A YANG Data Model for Routing Information Base (RIB)", draft-ietf-i2rs-rib-data-model-00 (work in progress), April 2015.

[I-D.ietf-i2rs-rib-info-model]
Bahadur, N., Folkes, R., Kini, S., and J. Medved, "Routing Information Base Info Model", draft-ietf-i2rs-rib-info-model-06 (work in progress), March 2015.

[I-D.ietf-i2rs-traceability] Clarke, J., Salgueiro, G., and C. Pignataro, "Interface to the Routing System (I2RS) Traceability: Framework and Information Model", draft-ietf-i2rs-traceability-03 (work in progress), May 2015.

[I-D.ietf-i2rs-usecase-reqs-summary]
Hares, S. and M. Chen, "Summary of I2RS Use Case Requirements", draft-ietf-i2rs-usecase-reqs-summary-01 (work in progress), May 2015.

[I-D.ietf-i2rs-yang-l2-network-topology]
Dong, J. and X. Wei, "A YANG Data Model for Layer-2 Network Topologies", draft-ietf-i2rs-yang-l2-network-topology-00 (work in progress), April 2015.

[I-D.ietf-i2rs-yang-network-topo]
Clemm, A., Medved, J., Varga, R., Tkacik, T., Bahadur, N., and H. Ananthakrishnan, "A Data Model for Network Topologies", draft-ietf-i2rs-yang-network-topo-01 (work in progress), June 2015.

[I-D.ietf-isis-yang-isis-cfg]
Litkowski, S., Yeung, D., Lindem, A., Zhang, J., and L. Lhotka, "YANG Data Model for ISIS protocol", draft-ietf-isis-yang-isis-cfg-02 (work in progress), March 2015.

[I-D.ietf-netconf-call-home] Watsen, K., "NETCONF Call Home and RESTCONF Call Home", draft-ietf-netconf-call-home-06 (work in progress), May 2015.

[I-D.ietf-netconf-restconf] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF

Protocol", draft-ietf-netconf-restconf-04 (work in progress), January 2015.

[I-D.ietf-netconf-restconf-collection] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Collection Resource", draft-ietf-netconf-restconf-collection-00 (work in progress), January 2015.

[I-D.ietf-netconf-zerotouch] Watsen, K., Clarke, J., and M. Abrahamsson, "Zero Touch Provisioning for NETCONF Call Home (ZeroTouch)", draft-ietf-netconf-zerotouch-02 (work in progress), March 2015.

[I-D.ietf-netmod-acl-model] Bogdanovic, D., Sreenivasa, K., Huang, L., and D. Blair, "Network Access Control List (ACL) YANG Data Model", draft-ietf-netmod-acl-model-02 (work in progress), March 2015.

[I-D.ietf-netmod-routing-cfg] Lhotka, L. and A. Lindem, "A YANG Data Model for Routing Management", draft-ietf-netmod-routing-cfg-19 (work in progress), May 2015.

[I-D.ietf-netmod-syslog-model] Wildes, C. and K. Sreenivasa, "SYSLOG YANG model", draft-ietf-netmod-syslog-model-03 (work in progress), March 2015.

[I-D.ietf-ospf-yang] Yeung, D., Qu, Y., Zhang, J., Bogdanovic, D., and K. Sreenivasa, "Yang Data Model for OSPF Protocol", draft-ietf-ospf-yang-00 (work in progress), March 2015.

[I-D.ietf-pcp-authentication] Wasserman, M., Hartman, S., Zhang, D., and T. Reddy, "Port Control Protocol (PCP) Authentication Mechanism", draft-ietf-pcp-authentication-09 (work in progress), May 2015.

[I-D.ietf-pcp-optimize-keepalives] Reddy, T., Patil, P., Isomaki, M., and D. Wing, "Optimizing NAT and

Firewall Keepalives Using Port Control Protocol (PCP)", draft-ietf-pcp-optimize-keepalives-06 (work in progress), May 2015.

[I-D.ietf-pcp-proxy] Perreault, S., Boucadair, M., Penno, R., Wing, D., and S. Cheshire, "Port Control Protocol (PCP) Proxy Function", draft-ietf-pcp-proxy-08 (work in progress), May 2015.

[I-D.ietf-sacm-architecture] Cam-Winget, N., Lorenzin, L., McDonald, I., and l. loxx@cisco.com, "Secure Automation and Continuous Monitoring (SACM) Architecture", draft-ietf-sacm-architecture-03 (work in progress), March 2015.

[I-D.ietf-sacm-terminology] Waltermire, D., Montville, A., Harrington, D., Cam-Winget, N., Lu, J., Ford, B., and M. Kaeo, "Terminology for Security Assessment", draft-ietf-sacm-terminology-06 (work in progress), February 2015.

[I-D.kini-i2rs-fb-rib-info-model] Kini, S., Hares, S., Ghanwani, A., Krishnan, R., Wu, Q., Bogdanovic, D., Tantsura, J., and R. White, "Filter-Based RIB Information Model", draft-kini-i2rs-fb-rib-info-model-00 (work in progress), March 2015.

[I-D.l3vpn-service-yang] Litkowski, S., Shakir, R., Tomotaki, L., and K. D'Souza, "YANG Data Model for L3VPN service delivery", draft-l3vpn- service-yang-00 (work in progress), February 2015.

[I-D.liu-bess-mvpn-yang] Liu, Y. and F. Guo, "Yang Data Model for Multicast in MPLS/BGP IP VPNs", draft-liu-bess-mvpn-yang-00 (work in progress), April 2015.

[I-D.shaikh-idr-bgp-model] Shaikh, A., D'Souza, K., Bansal, D., and R. Shakir, "BGP Model for Service Provider Networks", draft-shaikh-idr- bgp-model-01 (work in progress), March 2015.

[I-D.shaikh-rtgwg-policy-model] Shaikh, A., Shakir, R., D'Souza, K., and C. Chase, "Routing Policy Configuration Model for Service Provider Networks", draft-shaikh-rtgwg-policy-model-01 (work in progress), July 2015.

[I-D.tsingh-bess-pbb-evpn-yang-cfg] Tiruveedhula, K., Singh, T., Sajassi, A., Kumar, D., and L. Jalil, "YANG Data Model for PBB EVPN protocol", draft-tsingh-bess-pbb-evpn-yang-cfg-00 (work in progress), March 2015.

[I-D.zhang-i2rs-l1-topo-yang-model] Zhang, X., Rao, B., and X. Liu, "A YANG Data Model for Layer 1 Network Topology", draft-zhang-i2rs-l1-topo-yang-model-01 (work in progress), March 2015.

[I-D.zhdankin-idr-bgp-cfg] Alex, A., Patel, K., Clemm, A., Hares, S., Jethanandani, M., and X. Liu, "Yang Data Model for BGP Protocol", draft-zhdankin-idr-bgp-cfg-00 (work in progress), January 2015.

[I-D.zhuang-bess-evpn-yang] Zhuang, S. and Z. Li, "Yang Model for Ethernet VPN", draft-zhuang-bess-evpn-yang-00 (work in progress), December 2014.

[I-D.zhuang-bess-l3vpn-yang] Zhuang, S. and Z. Li, "Yang Data Model for BGP/MPLS IP VPNs", draft-zhuang-bess-l3vpn-yang-00 (work in progress), December 2014.

[RFC2748] Durham, D., Boyle, J., Cohen, R., Herzog, S., Rajan, R., and A. Sastry, "The COPS (Common Open Policy Service) Protocol", RFC 2748, January 2000.

[RFC2940] Smith, A., Partain, D., and J. Seligson, "Definitions of Managed Objects for Common Open Policy Service (COPS) Protocol Clients", RFC 2940, October 2000.

[RFC3084] Chan, K., Seligson, J., Durham, D., Gai, S.,

McCloghrie, K., Herzog, S., Reichmeyer, F., Yavatkar, R., and A. Smith, "COPS Usage for Policy Provisioning (COPS-PR)", RFC 3084, March 2001.

[RFC3303] Srisuresh, P., Kuthan, J., Rosenberg, J., Molitor, A., and A. Rayhan, "Middlebox communication architecture and framework", RFC 3303, August 2002.

[RFC3304] Swale, R., Mart, P., Sijben, P., Brim, S., and M. Shore, "Middlebox Communications (midcom) Protocol Requirements", RFC 3304, August 2002.

[RFC3483] Rawlins, D., Kulkarni, A., Bokaemper, M., and K. Chan, "Framework for Policy Usage Feedback for Common Open Policy Service with Policy Provisioning (COPS-PR)", RFC 3483, March 2003.

[RFC3484] Draves, R., "Default Address Selection for Internet Protocol version 6 (IPv6)", RFC 3484, February 2003.

[RFC4080] Hancock, R., Karagiannis, G., Loughney, J., and S. Van den Bosch, "Next Steps in Signaling (NSIS): Framework", RFC 4080, June 2005.

[RFC4261] Walker, J. and A. Kulkarni, "Common Open Policy Service (COPS) Over Transport Layer Security (TLS)", RFC 4261, December 2005.

[RFC4949] Shirey, R., "Internet Security Glossary, Version 2", RFC 4949, August 2007.

[RFC5189] Stiemerling, M., Quittek, J., and T. Taylor, "Middlebox Communication (MIDCOM) Protocol Semantics", RFC 5189, March 2008.

[RFC5277] Chisholm, S. and H. Trevino, "NETCONF Event Notifications", RFC 5277, July 2008.

[RFC5539] Badra, M., "NETCONF over Transport Layer Security (TLS)", RFC 5539, May 2009.

[RFC5973] Stiemerling, M., Tschofenig, H., Aoun, C., and E. Davies, "NAT/Firewall NSIS Signaling Layer Protocol (NSLP)", RFC 5973, October 2010.

[RFC6022] Scott, M. and M. Bjorklund, "YANG Module for NETCONF Monitoring", RFC 6022, October 2010.

[RFC6241] Enns, R., Bjorklund, M., Schoenwaelder, J., and A. Bierman, "Network Configuration Protocol (NETCONF)", RFC 6241, June 2011.

[RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", RFC 6242, June 2011.

[RFC6243] Bierman, A. and B. Lengyel, "With-defaults Capability for NETCONF", RFC 6243, June 2011.

[RFC6436] Amante, S., Carpenter, B., and S. Jiang, "Rationale for Update to the IPv6 Flow Label Specification", RFC 6436, November 2011.

[RFC6470] Bierman, A., "Network Configuration Protocol (NETCONF) Base Notifications", RFC 6470, February 2012.

[RFC6536] Bierman, A. and M. Bjorklund, "Network Configuration Protocol (NETCONF) Access Control Model", RFC 6536, March 2012.

[RFC6639] King, D. and M. Venkatesan, "Multiprotocol Label Switching Transport Profile (MPLS-TP) MIB-Based Management Overview", RFC 6639, June 2012.

[RFC6887] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, April 2013.

[RFC7223] Bjorklund, M., "A YANG Data Model for Interface Management", RFC 7223, May 2014.

[RFC7225] Boucadair, M., "Discovering NAT64 IPv6 Prefixes Using the Port Control Protocol (PCP)", RFC 7225, May 2014.

[RFC7277] Bjorklund, M., "A YANG Data Model for IP Management", RFC 7277, June 2014.

[RFC7317] Bierman, A. and M. Bjorklund, "A YANG Data Model for System Management", RFC 7317, August 2014.

Authors' Addresses

Susan Hares
Huawei
7453 Hickory Hill
Saline, MI 48176
USA
Email: shares@ndzh.com

Bob Moskowitz
HTT Consulting
Oak Park, MI 48237
Email: rgm@labs.htt-consult.com

Hosnieh Rafiee
http://www.rozanak.com
Munich, Germany
Phone: +49 (0) 17657587575
Email: ietf@rozanak.com

Dacheng Zhang
Beijing
China
Email: dacheng.zdc@aliabab-inc.com