

I2NSF  
Internet-Draft  
Intended status: Standards Track  
Expires: September 4, 2016

S. Hares  
J. Strassner  
Huawei  
D. Lopez  
Telefonica I+D  
L. Xia  
Huawei  
March 3, 2016

I2NSF Terminology  
draft-hares-i2nsf-terminology-00.txt

Abstract

This document describes the terminology for I2NSF.

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 September 4, 2016.

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 . . . . .	2
2. Terminology . . . . .	2
3. IANA Considerations . . . . .	8
4. Security Considerations . . . . .	8
5. References . . . . .	8
5.1. Normative References . . . . .	8
5.2. Informative References . . . . .	9
Authors' Addresses . . . . .	9

## 1. Introduction

This document describes the terminology for the work on the Interface to Security Functions (I2NSF). This section provides some background on I2NSF, but a problem statement can be found in [I-D.ietf-i2nsf-problem-and-use-cases]

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. 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 may consume network security services hosted by one or more providers, which may be their own enterprise, service providers, or a combination of both.

## 2. Terminology

AAA: Authentication, Authorization, and Accounting. See individual definitions.

**Abstraction:** An abstraction defines the salient characteristics and behavior of an object that distinguish it from all other types of objects. It manages complexity by exposing common properties between objects and processes while hiding detail that is not relevant.

**Accounting:** TBD

**ACL:** Access Control List. This is a mechanism for defining a set of permissions that are attached to an object.

**Action:** is a set of purposeful activities that have a set of associated behavior. (see I2NSF Action below.) (from [I-D.strassner-supra-generic-policy-info-model])

**Authentication:** TBD

**Authorization:** TBD

**B2B:** Business-to-Business

**Bespoke:** Something made to fit a particular person, client or company.

**Bespoke security management:** Security management systems which are made to fit a particular customer.

**Boolean Clause:** A logical statement that evaluates to either TRUE or FALSE. Also called Boolean Expression.

**Capability:** TBD

**Capability Layer:** TBD [Editorial comment from Strassner: the existing definition in use in documents is descriptive, not prescriptive.]

**Condition:** a set of attributes, features, and/or values that are to be compared with a set of known attributes, features, and/or values in order to make a decision. Examples of an I2NSF Condition include matching attributes of a packet or flow, and comparing the internal state of a NSF to a desired state. A Condition, when used in the context of a Policy Rule, is used to determine whether or not the set of Actions in that Policy Rule can be executed or not. (from [I-D.strassner-supra-generic-policy-info-model])

**Constraint:** A constraint is a limitation or restriction. Constraints may be added to any type of object (e.g., events, conditions, and actions in Policy Rules).

**Constraint Programming:** a type of programming that uses constraints to define relations between variables in order to find a feasible (and not necessarily optimal) solution.

**Context:** The Context of an Entity is a collection of measured and/or inferred knowledge that describe the state and the environment in which an Entity exists or has existed. (from <http://www.ietf.org/mail-archive/web/i2nsf/current/msg00762.html>)

**Controller:** TBD [Editorial: The definition is lacking content ("used interchangeably with Service Provider Security Controller or management system throughout this document") and overloaded - the two terms should be split into two separate definitions in documents.]

**DC:** Data Center

**Data Model:** A data model is a representation of concepts of interest to an environment in a form that is dependent on data repository, data definition language, query language, implementation language, and protocol (typically one or more of these ). (from [I-D.strassner-supra-generic-policy-info-model]).

**Event:** An Event is defined as any important occurrence in time of a change in the system being managed, and/or in the environment of the system being managed. Examples of an I2NSF Event include time and user actions (e.g. logon, logoff, and actions that violate and ACL.) An Event, when used in the context of a Policy Rule, is used to determine whether the condition clause of an imperative Policy Rule can be evaluated or not. (from [I-D.strassner-supra-generic-policy-info-model]).

**ECA:** Event - Condition - Action policy.

**FW:** Firewall

**Flow-based NSF:** A NSF 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, or NAT inspection or changes).

**I2NSF Action:** An I2NSF Action is a special type of Action that is used to control and monitor aspects of physical and virtual flow-based Network Security Functions. Examples of I2NSF Actions include providing intrusion detection and/or protection, web and flow filtering, and deep packet inspection for packets and flows. An I2NSF Action, when used in the context of a I2NSF Policy Rule, may be executed when both the event and the condition clauses of its owning I2NSF Policy Rule evaluate to true. The execution of this action may be influenced by applicable metadata. (see [I-D.strassner-supra-generic-policy-info-model]).

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

**I2NSF client:** A security client software component that utilizes the I2NSF protocol to read, write or change provisioning and operational aspects for the NSFs it attaches to by using the I2NSF protocol

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

**I2NSF Policy:** is a set of rules that are used to manage and control the changing or maintaining of the state of an security device.

**I2NSF Policy Rule:** is a policy rule that is adapted for I2NSF. The I2NSF Policy Rule is assumed to be in ECA form (i.e., an imperative structure). Other types of programming paradigms (e.g., declarative and functional) are currently out of scope. An example of an I2NSF Policy Rule is, in pseudo-code:

```
IF <event-clause> is TRUE
    IF <condition-clause> is TRUE
        THEN execute <action-clause>
    END-IF
END-IF
```

In the above example, the Event, Condition, and Action portions of a Policy Rule are all **\*\*Boolean Clauses\*\***.

I2NSF Registry: a registry which contains I2NSF capability information that can be controlled by the controller. (An expansion of Registry definition below.)

IDS: Intrusion Detection System (see below).

IPS: Intrusion Protection System (see below).

Information Model: An information model is a representation of concepts of interest to an environment in a form that is independent of data repository, data definition language, query language, implementation language, and protocol. (from [I-D.strassner-supra-generic-policy-info-model]).

Interface: is the set of operations one object knows it can invoke on another object. It is a subset of all operations that a given object implements. An example of multiple interfaces can be seen by considering the interfaces include a firewall uses. A firewall can have: a) multiple interfaces for data packets to traverse through and b) an interface for a controller to impose policy, or retrieve the results of execution of a policy rule. This illustrates that the same object may have multiple types of interfaces to serve different purposes.

Intrusion Detection System (IDS): a system which detects network intrusions via a variety of filters, monitors, and/or probes. An IDS may be stateful or stateless.

Intrusion Protection System (IPS): a system that protect against network intrusions. An IPS may be stateful or stateless.

Metadata: is data that provides information about other data. IETF network management protocols (e.g. NETCONF/RESTCONF/IPFix) or IETF routing interfaces (I2RS), and the I2NSF security interface may each utilize Metadata regarding the yang data models.

Middlebox: TBD

NSF: Network security function. An NSF is a function that that detects unwanted activity and blocks/mitigates the effect of such unwanted activity in order to support availability of a network. In addition, the NSF can help in supporting communication stream integrity and confidentiality.

OCL (the Object Constraint Language) is used to specify constraints in UML. (from <http://www.ietf.org/mail-archive/web/i2nsf/current/msg00762.html>)

OPNFV (Open Network Function Virtualization) TBD

**Policy Rule:** A Policy Rule is a set of rules that are used to manage and control the changing or maintaining of the state of one or more managed objects. Often this is shortened to Rule or Policy. (from [I-D.strassner-supra-generic-policy-info-model]). An I2NSF Policy Rule is assumed to be in ECA form (i.e., an imperative structure). Other types of programming paradigms (e.g., declarative and functional) are currently out of scope. For the complete definition of an I2NSF Policy Rule please see above. (see above I2NSF policy rule).

**Profile:** A structured representation of information that characterizes the capabilities of an object. This may be used to simplify how this object interacts with other objects in its environment. [Editors note: John Strassner suggestse this is a simplified defintion from a variety of sources (UAProf and CC/PP). It does not mention the concept of preference, therefore John wonders if we need a different definition here.]

**Registry:** is a logically centralized location containing data of a particular type; it may optionally contain metadata, relationships, and other aspects of the registered data in order to use those data effectively. An I2NSF registry is used to contain capability information that can be controlled by the controller.

**Registration Interface:** is an interface dedicated to requesting, receiving, editing, and deleting information in a Registry.

**Security Management System:** TBD (Editorial: Placeholder fro split of definition between controller (see above), and service provider security controller (see below) which existing I2NSF documents merge").

**Server Layer:** The Service Layer is called the Server Layer Interface in the I2NSF context.

**Service Layer:** The Service Layer (also called Client-Facing Interface) enables clients to manage security policies for their specific flows.

**Service Provider Security Controller:** TBD (Editorial: Place holder for a split between controller and security controller definition.)

**Tenant:** a tenant is a group of users that share common access privileges to the same software. An I2NSF tenant may be physical or virtual, and may run on a variety of systems or servers.

**Vendor Facing Interface:** The Vendor Facing Interface enables vendors to register their NSFs, along with the capabilities of their NSFs, with a logically centralized authority.

**Virtual NSF:** A NSF that is deployed as a distributed virtual device.

**Virtual Network Function (VNF):** A virtualized network component such as a router, switch, security box, or AAA Server.

**VNFM (VNF Manager):** Manager of virtual network functions that creates, deletes, manages, and moves VNFs.

**VNFPool:** a collection of interchangeable VNFs (i.e., each VNF has the same set of capabilities).

**Virtualization:** Virtualization is a type of software that creates a non-physical version of an object. Examples include virtualized operating systems, storage devices, and networking elements. [Editor's notes: Questions from John: Do we want or need to differentiate between different types of virtualization? For example: full vs. partial vs. para-virtualization (all types of "hardware virtualization")? Do we need to introduce OS virtualization? What about application virtualization?]

### 3. IANA Considerations

No IANA considerations exist for this document.

### 4. Security Considerations

This is a terminology document with no security considerations.

### 5. References

#### 5.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.

## 5.2. Informative References

- [I-D.ietf-i2nsf-gap-analysis]  
Hares, S., Moskowitz, R., and D. Zhang, "Analysis of Existing work for I2NSF", draft-ietf-i2nsf-gap-analysis-00 (work in progress), February 2016.
- [I-D.ietf-i2nsf-problem-and-use-cases]  
Hares, S., Dunbar, L., Lopez, D., Zarny, M., and C. Jacquenet, "I2NSF Problem Statement and Use cases", draft-ietf-i2nsf-problem-and-use-cases-00 (work in progress), February 2016.
- [I-D.ietf-netmod-acl-model]  
Bogdanovic, D., Koushik, K., Huang, L., and D. Blair, "Network Access Control List (ACL) YANG Data Model", draft-ietf-netmod-acl-model-06 (work in progress), December 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.
- [I-D.strassner-supra-generic-policy-info-model]  
Strassner, J., Halpern, J., and J. Coleman, "Generic Policy Information Model for Simplified Use of Policy Abstractions (SUPA)", draft-strassner-supra-generic-policy-info-model-04 (work in progress), February 2016.
- [RFC4948] Andersson, L., Davies, E., and L. Zhang, "Report from the IAB workshop on Unwanted Traffic March 9-10, 2006", RFC 4948, DOI 10.17487/RFC4948, August 2007, <<http://www.rfc-editor.org/info/rfc4948>>.
- [RFC7277] Bjorklund, M., "A YANG Data Model for IP Management", RFC 7277, DOI 10.17487/RFC7277, June 2014, <<http://www.rfc-editor.org/info/rfc7277>>.

## Authors' Addresses

Susan Hares  
Huawei  
7453 Hickory Hill  
Saline, MI 48176  
USA

Phone: +1-734-604-0332  
Email: shares@endzh.com

John Strassner  
Huawei  
Santa Clara, CA  
USA

Email: John.Strassner@huawei.com

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

Email: diego.r.lopez@telefonica.com

Liang Xia (Frank)  
Huawei  
101 Software Avenue, Yuhuatai District  
Nanjing , Jiangsu 210012  
China

Email: Frank.Xialiang@huawei.com