I2NSF S. Hares
Internet-Draft J. Strassner
Intended status: Informational Huawei
Expires: September 22, 2016 D. Lopez
Telefonica I+D
L. Xia
Huawei
March 21, 2016

Interface to Network Security Functions (I2NSF) Terminolgoy
draft-hares-i2nsf-terminology-01.txt

Abstract

This document defines a set of terms that are used for the Interface to Network Security Functions (I2NSF) effort.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

This document defines a set of terms that are used for the Interface to Security Functions(I2NSF) effort. This section provides some background on I2NSF, but a detailed problem statement can be found in [I-D.ietf-i2nsf-problem-and-use-cases]

The purposeof the document is to unify the terminology used among all the I2NSF documents.

Enterprises are now considering using network security functions (NSFs) hosted by service providers due to the growing challenges and complexity in maintaining a secure infrastructure, complying in maintaining an up to date secure infrastructure that complies with regulatory requirements while controlling costs. 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 (hosted) 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, but could also be provided to any kind of mass-market customers as well. 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.

It is out of scope of this document to define exhaustive lists of terms that are used in the security field in general; the reader is invited to refer to other documents such as [RFC4949]. [RFC4949] provides an excellent terminology glossary for the Internet Security Area is in [RFC4949].

The reader may also refer to [RFC3198] for a terminology document on policies (e.g., policy abstraction) and Policy-Based Management. The wise reader will have these documents at hand while using this terminology guide for I2NSF to provide additional answers.

2. Terminology

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

Abstraction: The definition of 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: The act of collecting information on resource usage for the purpose of trend analysis, auditing, billing, or cost allocation ([RFC2975] [RFC3539]

Access Control: Protection of system resources against unauthorized access; a process by which use of system resources is regulated according to a security policy and is permitted by only authorized entities (users, programs, processes, or other systems) according to that policy [RFC4949].

Acess Control List (ACL): This is a mechanism that implements access control for a system resource by enumerating the system entities that are permitted to access the resource and stating, either implicitly or explicitly, the access modes granted to each entity [RFC4949].

Action: Defines what is to be done when a set of conditions are met (See I2NSF Action). (from [I-D.strassner-supa-generic-policy-info-model])

Authentication: The act of verifying a claimed identity, in the form of a pre-existing label from a mutually known name space, as the originator of a message (message authentication) or as the end-point of a channel (entity authentication) [RFC3539].

Authorization: The act of determining if a particular right, such as access to some resource, can be granted to the presenter of a particular credential [RFC3539].

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

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

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

Capabilities: Defines a set of features that are available from a managed entity. (See also I2NSF Capability.)

Capability Layer: Defines an abstraction layer that exposes a set of capabilities of the I2NSF system.

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. 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. 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. (from [I-D.strassner-supa-generic-policy-info-model])

Constraint: A constraint is a limitation or restriction. Constraints may be associated with 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.]

Customer: A business role of an entity that is involved in the definition, consumption of services, and the possible negotiation of a contract to use services from a Provider.

Data Model: 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-supa-generic-policy-info-model]). [Editorial: this is This definition is different from that of RFC3198. See the referenced draft for specifics.]

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, traffic profile, and user actions (e.g. logon, logoff, and actions that violate an 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-supa-generic-policy-info-model]).

ECA: Event - Condition - Action policy.

Firewall (FW): Refers to a function that restricts data communication traffic to and from one of the connected networks (the one said to be "inside" the firewall) and thus protects that network’s system resources against threats from the other network (the one that is said to be "outside" the firewall) [RFC4949]. See also [I-D.ietf-opsawg-firewalls].

Flow-based NSF: A NSF that inspects network flows according to policies 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 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-supa-generic-policy-info-model]).

I2NSF Capability: Defines a set of features that are available from an NSF server.

I2NSF server: A software instance that implements a network security function that receives provisioning information and requests for operational data (e.g. monitoring data) from an I2NSF client. It is also responsible for enforcing the policies that it receives from an I2NSF client

I2NSF client: A software component that follows the I2NSF framework to read, write or change provisioning and operational aspects for the NSFs it attaches to.

I2NSF Management System: I2NSF client operates within a network management system, which serves as a collection and distribution point for I2NSF security provisioning and filtering of data.

I2NSF Policy: A set of rules that are used to manage and control the changing or maintaining of the state of an NSF instance.

I2NSF Policy Rule: 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 that contains I2NSF capability information that can be controlled by I2NSF Management System.

I2NSF System: Refers to the collection of I2NSF functional elements that contribute to provide the I2NSF service.

Information Model: 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-supa-generic-policy-info-model]).

Interface: Is the set of operations one object knows it can invoke (or expose to) 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: multiple interfaces for data packets to traverse through and an interface for a controller to impose policy, or retrieve the results of execution of a policy rule. The same object may have multiple types of interfaces to serve different (functional) 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: 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 to describe and/or prescribe characteristics and behavior of the YANG data models.

Middlebox: Is defined as any intermediary device performing functions other than the normal, standard functions of an IP router on the datagram path between a source host and destination host [RFC3234].

Network security function (NSF): Is a function that is provided as set of security-related service function. Typically, an NSF may be responsible for detecting unwanted activity and blocking/ mitigating the effect of such unwanted activity in order to fulfil the service requirements. The NSF can help in supporting communication stream integrity and confidentiality.

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

Policy Rule: 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-supa-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 objectin a given context. 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 definition 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: 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: An interface dedicated to requesting, receiving, editing, and deleting information in a registry.

Service Layer: The Service Layer (also called Client-Facing Interface) enables clients to manage security policies for their specific flows. [Editorial: Med suggest picking on eterm.]

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.

Editorial note on all Virtual functions: [MED] suggests removing virtual as the I2NSF does not make any assumptions about how things are created. Since this is a larger question - this section is left in with MED’s note.

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 Servier.

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, storagte devices, and networking elements. [Editoris notes: Questions from John: Do we want or need to differentiate between different tyeps 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. 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-supa-generic-policy-info-model]
Strassner, J., Halpern, J., and J. Coleman, "Generic Policy Information Model for Simplified Use of Policy Abstractions (SUPA)", draft-strassner-supa-generic-policy-info-model-04 (work in progress), February 2016.

[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>.

[RFC2975] Aboba, B., Arkko, J., and D. Harrington, "Introduction to Accounting Management", RFC 2975, DOI 10.17487/RFC2975, October 2000, <http://www.rfc-editor.org/info/rfc2975>.

[RFC3198] Westerinen, A., Schnizlein, J., Strassner, J., Scherling, M., Quinn, B., Herzog, S., Huynh, A., Carlson, M., Perry, J., and S. Waldbusser, "Terminology for Policy-Based Management", RFC 3198, DOI 10.17487/RFC3198, November 2001, <http://www.rfc-editor.org/info/rfc3198>.

[RFC3234] Carpenter, B. and S. Brim, "Middleboxes: Taxonomy and Issues", RFC 3234, DOI 10.17487/RFC3234, February 2002, <http://www.rfc-editor.org/info/rfc3234>.

[RFC3539] Aboba, B. and J. Wood, "Authentication, Authorization and Accounting (AAA) Transport Profile", RFC 3539, DOI 10.17487/RFC3539, June 2003, <http://www.rfc-editor.org/info/rfc3539>.

[RFC4949] Shirey, R., "Internet Security Glossary, Version 2", FYI 36, RFC 4949, DOI 10.17487/RFC4949, August 2007, <http://www.rfc-editor.org/info/rfc4949>.

[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@ndzh.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