I2NSF   S. Hares
Internet-Draft   J. Strassner
Intended status: Informational   Huawei
Expires: January 7, 2020   D. Lopez
                                                        Telefonica I+D
                                                               L. Xia
                                                               Huawei
                                                          H. Birkholz
                                                       Fraunhofer SIT
                                                        July 06, 2019

Interface to Network Security Functions (I2NSF) Terminology
draft-ietf-i2nsf-terminology-08

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 the terminology for the Interface to Network
   Security Functions (I2NSF) effort. This section provides some
   background on I2NSF; a detailed problem statement can be found in
   [RFC8192]. Motivation and comparison to previous work can be found
   in [I-D.ietf-i2nsf-gap-analysis].

   Enterprises are now considering using network security functions
   (NSFs) hosted by service providers due to the growing challenges and
   complexity 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
   mass-market customers as well. 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 in this document to define an exhaustive list of
   terms that are used in the security field; the reader is referred to
   other applicable documents, such as [RFC4949].

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 [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. Terminology

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

   Abstraction: The definition of 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.

   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 (e.g., users, programs, processes, or other
      systems) according to that policy [RFC4949].

   Access 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]. A YANG description is defined in
      [I-D.ietf-netmod-acl-model].

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

   Assertion: Defined by the ITU in [X.1252] as "a statement made by
      an entity without accompanying evidence of its validity". In the
      context of I2NSF, an assertion may include metadata about all or
      part of the assertion (e.g., context of the assertion, or about
      timestamp indicating the point in time the assertion was
      created). The validity of an assertion cannot be verified.
      (from [I-D.ietf-sacm-terminology]).

   Attestation: The process of validating the integrity of a computing
      device. See also Direct Anonymous Attestation, Remote Attestation.

Authentication: Defined in [RFC4949] as "the process of verifying
   a claim that a system entity or system resource has a certain
   attribute value." (from [I-D.ietf-sacm-terminology]).

Authorization: Defined in [RFC4949] as "an approval that is granted
   to a system entity to access a system resource."
   (from [I-D.ietf-sacm-terminology]).

Business-to-Business (B2B). A type of transaction in which one
   business makes a commercial transaction with another business.

Business-to-Consumer (B2C). A type of transaction in which a
   business makes a commercial transaction with a Customer.

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

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

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

Capability: A set of features that are available from an I2NSF
   Component. These features may, but do not have to, be used. All
   Capabilities are announced using the I2NSF Registration
   Interface. Capabilities are a type of I2NSF Metadata.

Component: An encapsulation of software that communicates using
   Interfaces. A Component may be implemented by hardware and/or
   software, and be represented using a set of classes. In general,
   a Component encapsulates a set of data structures and a set of
   algorithms that implement the function(s) that it provides.

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 (but 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: A Controller is a management Component that contains
   control plane functions to manage and facilitate information
   sharing. An I2NSF COntroller may also execute security functions.
   This definition is based on that in [I-D.ietf-sacm-terminology].

Control Plane: In the context of I2NSF, the Control Plane is an
   architectural Component that provides common control functions
   to all I2NSF Components, including some or all of the following:
   authentication, authorization, accounting, auditing, and
   Capability discovery and negotiation. The Control Plane
   orchestrates the operation of the Data Plane according to
   guidance and/or input from the Management Plane. I2NSF Components
   with Interfaces to the Control Plane may have knowledge of the
   Capabilities of other I2NSF Components within a particular I2NSF
   administrative domain. This definition is based on that in
   [I-D.ietf-sacm-terminology]. See also: Data Plane, Management
   Plane.

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

Data Center (DC): A facility used to house data processing and
   communication equipment.

Data Confidentiality: Defined in [RFC4949] as "the property that
   data is not disclosed to system entities unless they have been
   authorized to know the data."

Data Integrity: Defined in [RFC4949] as "the property that data has
   not been changed, destroyed, or lost in an unauthorized or
   accidental manner."

Data Model: 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). Note the difference
   between a data **model** and a data **structure**.
   See also: Information Model.

Data Plane: In the context of I2NSF, the Data Plane is an
   architectural Component that provides operational functions to
   enable an I2NSF Component to provide and consume packets and
   flows. See also: Control Plane, Management Plane.

Data Provenance: A historical record of the sources, origins and
   evolution of data that is influenced by inputs, entities,
   functions and processes.

   Data Structure: A low-level building block that is used in
      programming to implement an algorithm. It defines how data is
      organized. A data model typically contains multiple types of
      data structures; however, a data structure does not contain a
      data model. Note the difference between a data **model** and a
      data **structure**.

   Domain: A collection of Entities that share a common purpose. In
      addition, each constituent Entity in a Domain is both uniquely
      addressable and uniquely identifiable within that Domain.

   Direct Anonymous Attestation (DAA): A cryptographic primitive that
      enables remote authentication of a trusted computer without
      compromising the privacy of that computer's user(s). See also
      attestation, remote attestation.

   Firewall (FW): 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].
      [I-D.ietf-opsawg-firewalls]

   Flow: A set of information (e.g., packets) that are related in a
      fundamental manner (e.g., sent from the same source and sent to
      the same destination). A common example is a sequence of packets.
      It is the opposite of packet-based, which treats each packet
      discretely (e.g., each packet is assessed individually to
      determine the action(s) to be taken).

   Flow-based NSF: A NSF that inspects network flows according to a
      set of 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.

   I2NSF Action: An I2NSF Action is used to control and monitor
      aspects of flow-based NSFs. An I2NSF Action, when used in the
      context of an (imperative) I2NSF Policy Rule, may be executed
      only when the Event and the Condition clauses of its owning
      I2NSF Policy Rule evaluate to true. The execution of this I2NSF
      Action may be influenced by applicable metadata. Examples of
      I2NSF Actions include providing intrusion detection and/or
      protection, web and flow filtering, and deep packet inspection
      for packets and flows.
      See also: I2NSF Condition, I2NSF Event, I2NSF Policy Rule.

   I2NSF Agent: A software Component that implements an NSF. It
      typically plays the roles of I2NSF Consumer and I2NSF Producer.
      For example, it can receive provisioning information and requests
      for operational and/or monitoring data from an I2NSF Component,
      and can provide these and other data to I2NSF Consumers. It can
      also receive I2NSF Policy Rules to change the configuration of
      one or more network devices, optionally transform each I2NSF
      Policy Rule into an alternate form (e.g., one that is directly
      consummable by the network device), and then execute the I2NSF
      Policy Rules.

   I2NSF Component: A Component that provides one or more I2NSF
      Services. I2NSF Components are managed and communicate with other
      I2NSF Components using I2NSF Interfaces.

   I2NSF Condition: An I2NSF Condition is defined as 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
      determine whether or not the set of Actions in that (imperative)
      I2NSF Policy Rule can be executed or not. An I2NSF Condition,
      when used in the context of an (imperative) I2NSF Policy Rule,
      may be executed only when the Event clause of its owning
      I2NSF Policy Rule evaluates to true. Examples of an I2NSF
      Condition include matching attributes of a packet or flow, and
      comparing the internal state of an NSF to a desired state.
      See also: I2NSF Action, I2NSF Event, I2NSF Policy Rule.

   I2NSF Consumer: A Consumer is a Role that is assigned to an I2NSF
      Component that contains functions to request information from,
      and/or use services provided by, other I2NSF Components. Examples
      include requesting Capabilities and I2NSF Policy Rules from
      another I2NSF Component. See also: I2NSF Consumer-Facing
      Interface, I2NSF Producer, I2NSF Producer-Facing Interface, Role.

   I2NSF Consumer-Facing Interface: An Interface dedicated to
      requesting information from I2NSF Producers. This is typically
      defined per I2NSF administrative domain. For example, this
      Interface could be used to request a set of I2NSF Flow Security
      Policy Rules from an I2NSF Controller, or from one or more
      individual NSFs. See also: I2NSF Consumer, I2NSF Provider,
      I2NSF NSF-Facing Interface, Interface.

   I2NSF Directly Consummable Policy Rule: An I2NSF Policy Rule is
      said to be directly consummable if a network device can execute
      it without translating its content or structure. See also I2NSF
      Indirectly Consummable Policy Rule, I2NSF Policy Rule.

   I2NSF Indirectly Consummable Policy Rule: An I2NSF Policy Rule is

said to be indirectly consummable if a network device can NOT
execute it without first translating its content or structure. See
also I2NSF Directly Consummable Policy Rule, I2NSF Policy Rule.

   I2NSF Event: An I2NSF 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. An I2NSF Event, when used
      in the context of an (imperative) I2NSF Policy Rule, is used to
      determine whether the Condition clause of that Policy Rule can
      be evaluated or not. Examples of an I2NSF Event include time and
      user actions (e.g. logon, logoff, and actions that violate an
      ACL). See also I2NSF Action, I2NSF Condition, I2NSF Policy Rule.

    I2NSF Management System: I2NSF Consumers and Producers operate
      within the scope of a network management system, which serves as
      a collection and distribution point for I2NSF security
      provisioning, monitoring, and other operations.

   I2NSF NSF-Facing Interface: An Interface dedicated to providing I2NSF
      Services. For example, this could provide Anti-Virus, (D)DoS, or
      IPS Services. This is also called the "NSF-Facing Interface".
      See also: Interface, I2NSF Consumer Interface.

   I2NSF Policy Rule: An I2NSF Policy Rule is an imperative statement
      that is used as a means to monitor and control the changing and/or
      maintaining of the state of one or more managed objects. It
      consists of three clauses (Event, Condition, and Action). The
      Event and Condition clauses are Boolean clauses, while the Action
      clause consists of a set of one or more I2NSF Actions.
      In this context, "manage" means that one or more of the following
      six fundamental operations are supported: create, read, write,
      delete, start, and stop. Note that for this release of I2NSF,
      only imperative policy rules are in 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

    I2NSF Producer: A Producer is a Role that is assigned to an I2NSF
      Component that contains functions to send information, and/or
      provide services, to another I2NSF Component (e.g., for
      describing, communicating, and/or executing policies, or for
      transmitting data and/or metadata). See also: I2NSF Consumer,
      I2NSF Consumer-Facing Interface, I2NSF Producer, I2NSF
      Producer-Facing Interface, Role.

   I2NSF Registry: A repository where I2NSF data and metadata
      information (e.g., alarms and Capabilities, respectively) are
      stored and maintained. I2NSF Components can connect to the I2NSF
      Registry using the I2NSF Registration Interface.
      Operations performed on an I2NSF Registry SHOULD be defined using
      an Access Control mechanism. Examples of information that SHOULD
      be registered include Capability metdata, as well as consistent
      defintions of data and I2NSF Components.
      See also: Access Control, I2NSF Component, I2NSF Consumer,
      I2NSF Provider, I2NSF Registration Interface.

   I2NSF Registration Interface: An Interface dedicated to
      requesting information from, and writing information about,
      I2NSF Components and Capabilities. See also: I2NSF Component,
      I2NSF Consumer, I2NSF Provider, I2NSF Registry.

   I2NSF Service: A set of functions, provided by an I2NSF Component,
      which provides data communication, processing, storage,
      presentation, manipulation, or other functions that can be
      consumed by I2NSF Components. Exemplary I2NSF Services include
      Anti-Virus, Authentication, Authorization, Firewall, and IPS
      Services. See also: I2NSF Component, Interface.

   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. See also: Data Model.

   Interface: A set of operations one object knows it can invoke on,
      and expose to, another object. It is a subset of all operations
      that a given object implements. The same object may have multiple
      types of interfaces to serve different purposes.
      See also: I2NSF Component, I2NSF Consumer-Facing Interface, I2NSF
      Registration Interface, Interface Group, NSF-Facing Interface

   Interface Group: A set of Interfaces that are related in purpose and
      which share the same communication mechanisms.
      See also: Interface.

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

   Intrusion Protection System (IPS): A system that protects against
      network intrusions. An IPS may be stateful or stateless.
      See also: IDS.

   Management Domain: A collection of Entities that share a common
      purpose, which has the following three behavioral features:
         1) a set of administrators are assigned to govern the Entities
            that are contained in a Management Domain
         2) a set of applications are defined that are responsible for
            executing one or more governance operations
         3) a set of management mechanisms, such as Policy Rules, are
            defined to govern the behavior of the Entities contained
            in the Management Domain.

   Management Plane: In the context of I2NSF, the Management Plane is
      an architectural Component that provides common functions to
      define the behavior of I2NSF Components. The primary use of the
      Management Plane is to formulate behavioral commands and forward
      them to the Control Plane. The Control Plane then translates them
      into a form that can be consumed by I2NSF components. The
      Management Plane may also instantiate and manage I2NSF Policy
      Rules. The Management Plane is also responsible for handling and
      acting on OAM data, which may influence the decision-making
      processes in the I2NSF Control Plane and other I2NSF Components.
      See also: Control Plane, Data Plane.

   Metadata: Data that provides information about other data.
      Examples include IETF network management protocols (e.g. NETCONF,
      RESTCONF, IPFIX) or IETF routing interfaces (I2RS). The I2NSF
      security interface may utilize Metadata to describe and/or
      prescribe characteristics and behavior of the YANG data models.

   Middlebox: 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): Software that provides a set of
      security-related services. Examples include detecting unwanted
      activity and blocking or mitigating the effect of such unwanted
      activity in order to fulfil service requirements. The NSF can
      also help in supporting communication stream integrity and
      confidentiality.

   NSF-Facing Interface: An Interface dedicated to specifying and
      monitoring I2NSF Policy Rules that are enforced by one or more
      NSFs. This is typically defined per I2NSF administrative
      domain. Note that all features of a given NSF do not have to be
      used. See also: Consumer-Facing Interface, Interface.

   Object Constraint Language (OCL): A constraint programming language
      that is used to specify restrictions on functionality. (from

http://www.ietf.org/mail-archive/web/i2nsf/current/msg00762.html)

   Profile: A structured representation of information that uses a
      pre-defined set of capabilities of an object, typically in a
      specific context. Zero or more Capabilities may be changed at
      runtime. This may be used to simplify how this object interacts
      with other objects in its environment.

   Remote Attestation: A function that enables changes to an Entity to
      be detected by authorized parties (e.g., applications or users).
      Direct Anonymous Attestation preserves the privacy of the user,
      whereas remote attestation may not. See also: Attestation,
      Direct Anonymous Attestation.

   Role: An abstraction of a Component that models context-specific
      views and responsibilities of an object as separate Role objects.
      Role objects can optionally be attached to, and removed from, the
      object that the Role object describes at runtime. This provides
      three important benefits. First, it enables different behavior
      to be supported by the same Component for different contexts.
      Second, it enables the behavior of a Component to be adjusted
      dynamically (i.e., at runtime, in response to changes in context)
      by using one or more Roles to define the behavior desired for
      each context. Third, it decouples the Roles of a Component from
      the Applications that use the Component.

   Tenant: 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.

3. IANA Considerations

   No IANA considerations exist for this document.

4. Security Considerations

   This is a terminology document with no security considerations.

5. Contributors

   The following people contributed to creating this document, and are
   listed in alphabetical order:

      Adrian Farrel, Christian Jacquenet, Linda Dunbar,
      Mohammed Boucadair

6. References

6.1. Informative References

   [I-D.ietf-i2nsf-gap-analysis]
             Hares, S., Moskowitz, R., and Zhang, D., "Analysis of
             Existing work for I2NSF", draft-ietf-i2nsf-gap-analysis-03
             (work in progress), March 2017.

   [RFC8192]
             Hares, S., Dunbar, L., Lopez, D., Zarny, M., and C.
             Jacquenet, "I2NSF Problem Statement and Use cases",
             RFC 8192, July 2017.

   [I-D.ietf-netmod-acl-model]
             Bogdanovic, D., Sreenivasa, K., Huang, L., Blair, D.,
             "Network Access Control List (ACL) YANG Data Model",
             draft-ietf-netmod-acl-model-14 (work in progress),
             October 2017.

   [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.ietf-sacm-terminology]
             Birkholz, H., Lu, J., Strassner, J., Cam-Wignet, N.,
             "Secure Automation and Continuous Monitoring (SACM)
             Terminology", draft-ietf-sacm-terminology-14,
             December 2017.

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

   [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>.

   [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>.

   [X.1252]   ITU-T, "Baseline identity management terms and
             definitions", Recommendation ITU-T X.1252, April 2510

Authors' Addresses

   Susan Hares
   Huawei
   7453 Hickory Hill
   Saline, MI   USA   48176
   Phone: +1-734-604-0332
   Email: shares@ndzh.com

   John Strassner
   Huawei Technologies
   Santa Clara, CA   USA   95050
   Email: john.sc.strassner@huawei.com

   Diego R. Lopez
   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

   Henk Birkholz
   Fraunhofer SIT
   Rheinstrasse 75
   Darmstadt   64295
   Germany
   Email: henk.birkholz@sit.fraunhofer.de