Workgroup: I2NSF Working Group
Internet-Draft:
draft-ietf-i2nsf-consumer-facing-interface-dm-15
Published: 15 September 2021
Intended Status: Standards Track
Expires: 19 March 2022
Authors: J. Jeong, Ed. C. Chung
Sungkyunkwan University Sungkyunkwan University
T. Ahn R. Kumar S. Hares
Korea Telecom Juniper Networks Huawei

I2NSF Consumer-Facing Interface YANG Data Model

Abstract

This document describes an information model and a YANG data model for the Consumer-Facing Interface between an Interface to Network Security Functions (I2NSF) User and Security Controller in an I2NSF system in a Network Functions Virtualization (NFV) environment. The information model defines various types of managed objects and the relationship among them needed to build the interface. The information model is based on the "Event-Condition-Action" (ECA) policy model defined by a capability information model for I2NSF, and the data model is defined for enabling different users of a given I2NSF system to define, manage, and monitor security policies for specific flows within an administrative domain.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction
2. Terminology
3. Information Model for Policy
  3.1. Event Sub-model
  3.2. Condition Sub-model
  3.3. Action Sub-model
4. Information Model for Policy Endpoint Groups
  4.1. User Group
  4.2. Device Group
  4.3. Location Group
  4.4. URL Group
5. Information Model for Threat Prevention
  5.1. Threat Feed
  5.2. Payload Content
6. Network Configuration Access Control Model (NACM) for I2NSF Consumer-Facing Interface
7. YANG Data Model of Consumer-Facing Interface
  7.1. YANG Module of Consumer-Facing Interface
8. XML Configuration Examples of High-Level Security Policy Rules
  8.1. Database Registration: Information of Positions and Devices (Endpoint Group)
  8.2. Scenario 1: Block SNS Access during Business Hours
  8.3. Scenario 2: Block Malicious VoIP/VoLTE Packets Coming to a Company
  8.4. Scenario 3: Mitigate HTTP and HTTPS Flood Attacks on a Company Web Server
9. XML Configuration Example of a User Group's Access Control for I2NSF Consumer-Facing Interface
10. IANA Considerations
11. Security Considerations
12. Acknowledgments
13. Contributors
14. References
  14.1. Normative References
  14.2. Informative References
Appendix A. Changes from draft-ietf-i2nsf-consumer-facing-interface-dm-14

Authors' Addresses

1. Introduction

In a framework of Interface to Network Security Functions (I2NSF) [RFC8329], each vendor can register their NSFs using a Developer's Management System (DMS). Assuming that vendors also provide the front-end web applications registered with an I2NSF User, the Consumer-Facing Interface is required because the web applications developed by each vendor need to have a standard interface specifying the data types used when the I2NSF User and Security Controller communicate using this interface. Therefore, this document specifies the required information, their data types, and encoding schemes so that high-level security policies (or configuration information for security policies) can be transferred to the Security Controller through the Consumer-Facing Interface. These policies can easily be translated by the Security Controller into low-level security policies. The Security Controller delivers the translated policies to Network Security Functions (NSFs) according to their respective security capabilities for the required securiy enforcement.

The Consumer-Facing Interface would be built using a set of objects, with each object capturing a unique set of information from Security Administrator (i.e., I2NSF User [RFC8329]) needed to express a Security Policy. An object may have relationship with various other objects to express a complete set of requirements. An information model captures the managed objects and relationship among these objects. The information model proposed in this document is structured in accordance with the "Event-Condition-Action" (ECA) policy model.

An NSF Capability model is proposed in [I-D.ietf-i2nsf-capability] as the basic model for both the NSF-Facing interface and Consumer-Facing Interface security policy model of this document.

[RFC3444] explains differences between an information and data model. This document uses the guidelines in [RFC3444] to define both the information and data model for Consumer-Facing Interface. Figure 1 shows a high-level abstraction of Consumer-Facing Interface. A data model, which represents an implementation of the information model in a specific data representation language, is also defined in this document.

```
                 +-----------------+
                 | Consumer-Facing |
                 |    Interface    |
                 +--------+--------+
                          ^
                          |
                          +-------------+------------+
                          |             |            |
                    +-----+----+  +-----+----+  +----+---+
                    |  Policy  |  | Endpoint |  | Threat |
                    |          |  |  groups  |  |  feed  |
                    +-----+----+  +----------+  +--------+
                          ^
                          |
                   +------+------+
                   |     Rule    |
                   +------+------+
                          ^
                          |
         +----------------+----------------+
         |                |                |
  +------+------+  +------+------+  +------+------+
  |    Event    |  |  Condition  |  |    Action   |
  +-------------+  +-------------+  +-------------+
```

Figure 1: Diagram for High-level Abstraction of Consumer-Facing Interface

Data models are defined at a lower level of abstraction and provide many details. They provide details about the implementation of a protocol's specification, e.g., rules that explain how to map managed objects onto lower-level protocol constructs. Since conceptual models can be implemented in different ways, multiple data models can be derived from a single information model.

The efficient and flexible provisioning of network functions by a Network Functions Virtualization (NFV) system leads to a rapid advance in the network industry. As practical applications, Network Security Functions (NSFs), such as firewall, Intrusion Detection System (IDS)/Intrusion Prevention System (IPS), and attack mitigation, can also be provided as Virtual Network Functions (VNF) in the NFV system. By the efficient virtualization technology, these VNFs might be automatically provisioned and dynamically migrated based on real-time security requirements. This document presents a YANG data model to implement security functions based on NFV.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

This document uses the terminology described in [RFC8329].

This document follows the guidelines of [RFC8407], uses the common YANG types defined in [RFC6991], and adopts the Network Management Datastore Architecture (NMDA). The meaning of the symbols in tree diagrams is defined in [RFC8340].

3. Information Model for Policy

A Policy object represents a mechanism to express a Security Policy by Security Administrator (i.e., I2NSF User) using Consumer-Facing Interface toward Security Controller; the policy would be enforced on an NSF. Figure 2 shows the YANG tree of the Policy object. The Policy object SHALL have the following information:

Name: This field identifies the name of this object.

Resolution-strategy: This field represent how to resolve conflicts that occur between actions of the same or different policy rules that are matched and contained in this particular NSF.

Rules: This field contains a list of rules. These rules are defined for 1) communication between two Endpoint Groups, 2) for preventing communication with externally or internally identified threats, and 3) for implementing business requirement such as controlling access to internal or external resources for meeting regulatory compliance or business objectives. An organization may restrict certain communication between a set of user and applications for example. The threats may be from threat feeds obtained from external sources or dynamically identified by using specialty devices in the network. Rule conflict analysis should be triggered by the monitoring service to perform an exhaustive detection of anomalies among the configuration rules installed into the security functions.

```
+--rw i2nsf-cfi-policy* [policy-name]
   +--rw policy-name            string
   +--rw resolution-strategy?   identityref
   +--rw rules* [rule-name]
   |  ...
   +--rw endpoint-groups
   |  ...
   +--rw threat-preventions
   |  ...
   +--rw url-group* [name]
   |  ...
```

Figure 2: Policy YANG Data Tree

A policy is a list of rules. In order to express a Rule, a Rule must have complete information such as where and when a policy needs to be applied. This is done by defining a set of managed objects and relationship among them. A Policy Rule may be related segmentation, threat mitigation or telemetry data collection from an NSF in the network, which will be specified as the sub-model of the policy model in the subsequent sections. Figure 3 shows the YANG data tree of the Rule object. The rule object SHALL have the following information:

Rule-Name: This field identifies the name of this object.

Priority: This field identifies the priority of the rule.

Event: This field includes the information to determine whether the Rule Condition can be evaluated or not. See details in Section 4.1.

Condition: This field contains all the checking conditions to apply to the objective traffic. See details in Section 4.2.

Action: This field identifies the action taken when a rule is matched. There is always an implicit action to drop traffic if no rule is matched for a traffic type. See details in Section 4.3.

```
  +--rw rules* [rule-name]
  |  +--rw rule-name    string
  |  +--rw priority?    uint8
  |  +--rw event
  |  ...
  |  +--rw condition
  |  ...
  |  +--rw actions
     ...
```

Figure 3: Rule YANG Data Tree

Note that in the case of policy conflicts, the resolution of the conflicted policies conforms to the guidelines of "Information Model of NSFs Capabilities" [I-D.ietf-i2nsf-capability].

3.1. Event Sub-model

The Event Object contains information related to scheduling a Rule. The Rule could be activated based on a set time or security event. Figure 4 shows the YANG tree of the Event object. Event object SHALL have following information:

Security-event: This field identifies for which security event the policy is enforced. The examples of security events are: "DDOS", "spyware", "trojan", and "ransomware".

Time-information: This represents the security rule is enforced based on the period information with the end time for the event.

Start-date-time: This represents the start time of the event. The rule will start repeating from the specified time"

End-date-time: This represents the end time of the event. If the rule time has pass the end-time, the rule will stop repeating"

Period: This represents the period of time the rule event is active. It can be configured by the start-time, stop-time, day, date, and month.

Frequency: This represents how frequent the rule should be enforced. There are four options: "only-once", "daily", "weekly", "monthly" or "yearly".

```
+--rw event
|  +--rw security-event?   identityref
|  +--rw time
|     +--rw start-date-time?   yang:date-and-time
|     +--rw end-date-time?     yang:date-and-time
|     +--rw period
|     |  +--rw start-time?   time
|     |  +--rw end-time?     time
|     |  +--rw day*          identityref
|     |  +--rw date*         int32
|     |  +--rw month*        string
|     +--rw frequency?         enumeration
```

Figure 4: Event Sub-model YANG Data Tree

3.2. Condition Sub-model

This object represents Conditions that Security Administrator wants to apply the checking on the traffic in order to determine whether the set of actions in the Rule can be executed or not. The Condition Sub-model consists of three different types of containers each representing different cases, such as general firewall and DDoS-mitigation cases, and a case when the condition is based on the payload strings of packets. Each containers have source and destination-target to represent the source and destination for each case. Figure 5 shows the YANG tree of the Condition object. The Condition Sub-model SHALL have following information:

Case (firewall-condition): This field represents the general firewall case, where a security admin can set up firewall conditions using the information present in this field. The source and destination is represented as source, destination, transport layer protocol, port numbers, and ICMP parameters.

Case (ddos-condition): This field represents the condition for DDoS mitigation, where a security admin can set up DDoS mitigation conditions using the information present in this field. The rate of packet, byte, or flow threshold can be configured to mitigate the DDoS.

Case (anti-virus-condition): This field represents the condition for Antivirus, where a security admin can set up Antivirus conditions using the information present in this field. The file names or types can be configured to be allowed without the Antivirus interuption.

Case (payload-condition): This field contains the payload string information. This information is useful when security rule condition is based on the string contents of incoming or outgoing packets. The name referring to the payload-groups defined and registered in the endpoint-groups.

Case (url-condition): This field represents the URL to be filtered. This information can be used to block or allow a certain URL or website. The url-name is a group of URL or websites to be matched.

Case (voice-condition): This field contains the call source-id, call destination-id, and user-agent. This information can be used to filter a caller id or receiver id to prevent any VoIP or VoLTE exploits or attack.

Case (context-condition): This field represents a context of a packet or flow. The context can be extended. This module provides a context of geography location.

Case (Threat-feed-condition): This field contains the information obtained from threat-feeds (e.g., Palo-Alto, or RSA-netwitness). This information is useful when security rule condition is based on the existing threat reports gathered by other sources.

```
  +--rw condition
  |  +--rw firewall-condition
  |  |  +--rw source*                     union
  |  |  +--rw destination*                union
  |  |  +--rw transport-layer-protocol?   identityref
  |  |  +--rw range-port-number
  |  |  |  +--rw start-port-number?   inet:port-number
  |  |  |  +--rw end-port-number?     inet:port-number
  |  |  +--rw icmp* [version]
  |  |     +--rw version    enumeration
  |  |     +--rw type*      uint8
  |  |     +--rw code*      uint8
  |  +--rw ddos-condition
  |  |  +--rw rate-limit
  |  |     +--rw packet-rate-threshold?   uint32
  |  |     +--rw byte-rate-threshold?     uint32
  |  |     +--rw flow-rate-threshold?     uint32
  |  +--rw anti-virus-condition
  |  |  +--rw exception-files*   string
  |  +--rw payload-condition
  |  |  +--rw content*
         -> /i2nsf-cfi-policy/threat-preventions/payload-content/name
  |  +--rw url-condition
  |  |  +--rw url-name?
         -> /i2nsf-cfi-policy/endpoint-groups/url-group/name
  |  +--rw voice-condition
  |  |  +--rw source-id*        string
  |  |  +--rw destination-id*   string
  |  |  +--rw user-agent*       string
  |  +--rw context-condition
  |     +--rw geography-location-condition
  |        +--rw source*
            -> /i2nsf-cfi-policy/endpoint-groups/location-group/name
  |        +--rw destination*
            -> /i2nsf-cfi-policy/endpoint-groups/location-group/name
  |  |  +--rw threat-feed-condition
  |  |     +--rw name*
        -> /i2nsf-cfi-policy/threat-preventions/threat-feed-list/name
```

Figure 5: Condition Sub-model YANG Data Tree

3.3. Action Sub-model

This object represents actions that Security Admin wants to perform based on certain traffic class. Figure 6 shows the YANG tree of the Action object. The Action object SHALL have following information:

Primary-action:

This field identifies the action when a rule is matched by an NSF. The action could be one of "pass", "drop", "rate-limit", "mirror", "invoke-signaling", "tunnel-encapsulation", "forwarding", and "transformation".

Secondary-action: This field identifies the action when a rule is matched by an NSF. The action could be one of "rule-log" and "session-log".

```
        +--rw actions
        |  +--rw primary-action
        |  |  +--rw action?   identityref
        |  +--rw secondary-action
        |     +--rw log-action?   identityref
```

Figure 6: Action Sub-model YANG Data Tree

4. Information Model for Policy Endpoint Groups

The Policy Endpoint Group is a very important part of building User-Construct based policies. A Security Administrator would create and use these objects to represent a logical entity in their business environment, where a Security Policy is to be applied. There are multiple managed objects that constitute a Policy's Endpoint Group, as shown in Figure 7. Figure 8 shows the YANG tree of the Endpoint-Groups object. This section lists these objects and relationship among them.

It is assumed that the information of Endpoint Groups (e.g., User-group, Device-group, and Location-group) such as the IP address(es) of each member in a group are stored in the I2NSF database available to the Security Controller, and that the IP address information of each group in the I2NSF database is synchronized with other systems in the networks under the same administration.

```
                        +--------------------+
                        |  Endpoint Groups   |
                        +---------+----------+
                                  ^
                                  |
           +--------------+-------+--------+---------------+
    0..n   |      0..n    |       0..n     |          0..n |
     +-----+----+  +------+-----+  +-------+------+  +-----+---+
     |User-group|  |Device-group|  |Location-group|  |Url-group|
     +----------+  +------------+  +--------------+  +---------+
```

Figure 7: Endpoint Group Diagram

```
+--rw endpoint-groups
|  +--rw user-group* [name]
|  ...
|  +--rw device-group* [name]
|  ...
|  +--rw location-group* [name]
|  ...
|  +--rw url-group* [name]
|  ...
```

Figure 8: Endpoint Group YANG Data Tree

4.1. User Group

This object represents a User-Group. Figure 9 shows the YANG tree of the User-Group object. The User-Group object SHALL have the following information:

Name: This field identifies the name of this object.

mac-address: This represents the MAC address of a user in the user group.

Range-ipv4-address: This represents the IPv4 address range of a user in the user group.

Range-ipv6-address: This represents the IPv6 address range of a user in the user group.

```
+--rw user-group* [name]
|  +--rw name                              string
|  +--rw mac-address*                      yang:mac-address
|  +--rw (match-type)
|  |  +--:(range-match-ipv4)
|  |  |  +--rw range-ipv4-address
|  |  |     +--rw start-ipv4-address    inet:ipv4-address-no-zone
|  |  |     +--rw end-ipv4-address      inet:ipv4-address-no-zone
|  |  +--:(range-match-ipv6)
|  |     +--rw range-ipv6-address
|  |        +--rw start-ipv6-address    inet:ipv6-address-no-zone
|  |        +--rw end-ipv6-address      inet:ipv6-address-no-zone
```

Figure 9: User Group YANG Data Tree

4.2. Device Group

This object represents a Device-Group. Figure 10 shows the YANG tree of the Device-group object. The Device-Group object SHALL have the following information:

Name: This field identifies the name of this object.

IPv4: This represents the IPv4 address of a device in the device group.

IPv6: This represents the IPv6 address of a device in the device group.

Range-ipv4-address: This represents the IPv4 address range of a device in the device group.

Range-ipv6-address: This represents the IPv6 address range of a device in the device group.

Application-protocol: This represents the application layer protocols of devices. If this is not set, it cannot support the appropriate protocol

```
+--rw device-group* [name]
|  +--rw name                        string
|  +--rw (match-type)
|  |  +--:(range-match-ipv4)
|  |  |  +--rw range-ipv4-address
|  |  |     +--rw start-ipv4-address    inet:ipv4-address-no-zone
|  |  |     +--rw end-ipv4-address      inet:ipv4-address-no-zone
|  |  +--:(range-match-ipv6)
|  |     +--rw range-ipv6-address
|  |        +--rw start-ipv6-address    inet:ipv6-address-no-zone
|  |        +--rw end-ipv6-address      inet:ipv6-address-no-zone
|  +--rw application-protocol*       identityref
```

Figure 10: Device Group YANG Data Tree

4.3. Location Group

This object represents a location group based on either tag or other information. Figure 11 shows the YANG tree of the Location-Group object. The Location-Group object SHALL have the following information:

Name: This field identifies the name of this object.

Geo-ip-ipv4: This field represents the IPv4 Geo-ip address of a location [RFC8805].

Geo-ip-ipv6: This field represents the IPv6 Geo-ip address of a location [RFC8805].

Continent: This field represents the continent where the location group member is located.

```
+--rw location-group* [name]
|  +--rw name            string
|  +--rw geo-ip-ipv4* [ipv4-address]
|  |  +--rw ipv4-address    inet:ipv4-address-no-zone
|  |  +--rw ipv4-prefix?    inet:ipv4-prefix
|  +--rw geo-ip-ipv6* [ipv6-address]
|  |  +--rw ipv6-address    inet:ipv6-address-no-zone
|  |  +--rw ipv6-prefix?    inet:ipv6-prefix
|  +--rw continent?     identityref
```

Figure 11: Location Group YANG Data Tree

4.4. URL Group

This object represents a URL group based on a Uniform Resource Locator (URL) or web address. Figure 12 shows the YANG tree of the URL-Group object. The URLn-Group object SHALL have the following information:

Name: This field identifies the name of this object.

url: This field represents the new URL added by a user to the URL database.

```
+--rw url-group* [name]
   +--rw name             string
   +--rw url*   string
```

Figure 12: URL Group YANG Data Tree

5. Information Model for Threat Prevention

The threat prevention plays an important part in the overall security posture by reducing the attack surfaces. This information could come from various threat feeds (i.e., sources for obtaining the threat information). There are multiple managed objects that constitute this category. This section lists these objects and relationship among them. Figure 14 shows the YANG tree of a Threat-Prevention object.

```
                +-------------------+
                | Threat Prevention |
                +---------+---------+
                          ^
                          |
                +---------+---------+
        0..n    |         0..n      |
         +------+------+   +--------+--------+
         | Threat-feed |   | payload-content |
         +-------------+   +-----------------+
```

Figure 13: Threat Prevention Diagram

```
     +--rw threat-prevention
        +--rw threat-feed-list* [name]
           ...
        +--rw payload-content* [name]
           ...
```

Figure 14: Threat Prevention YANG Data Tree

5.1. Threat Feed

This object represents a threat feed which provides the signatures of malicious activities. Figure 15 shows the YANG tree of a Threat-feed-list. The Threat-Feed object SHALL have the following information:

Name: This field identifies the name of this object.

Description: This is the description of the threat feed. The description should have the clear indication of the security attack such as attack type (e.g., APT) and file types used (e.g., executable malware).

Signatures: This field contains the threat signatures of malicious programs or activities provided by the threat-feed. The examples of signature types are "YARA", "SURICATA", and "SNORT" [YARA] [SURICATA][SNORT].

It is assumed that the I2NSF User obtains the threat signatures (i.e., threat content patterns) from a threat-feed server (i.e., feed provider), which is a server providing threat signatures. With the obtained threat signatures, the I2NSF User can deliver them to the Security Controller. The retrieval of the threat signatures by the I2NSF User is out of scope in this document.

```
       +--rw threat-prevention
          +--rw threat-feed-list* [name]
             +--rw name                identityref
             +--rw description?        string
             +--rw signatures*              identityref
```

Figure 15: Threat Feed YANG Data Tree

5.2. Payload Content

This object represents a custom list created for the purpose of defining an exception to threat feeds. Figure 16 shows the YANG tree of a Payload-content list. The Payload-Content object SHALL have the following information:

Name: This field identifies the name of this object. For example, the name "backdoor" indicates the payload content is related to a backdoor attack.

Description: This represents the description of how the payload content is related to a security attack.

Content: This contains the payload contents, which are involed in a security attack, such as strings.

```
       +--rw payload-content* [name]
          +--rw name         string
          +--rw description  string
          +--rw content*     string
```

Figure 16: Payload Content in YANG Data Tree

6. Network Configuration Access Control Model (NACM) for I2NSF Consumer-Facing Interface

Network Configuration Access Control Model (NACM) provides a user group with an access control with the following features [RFC8341]:

*Independent control of action, data, and notification access is provided.

*A simple and familiar set of datastore permissions is used.

- Support for YANG security tagging allows default security modes to automatically exclude sensitive data.

- Separate default access modes for read, write, and execute permissions are provided.

- Access control rules are applied to configurable groups of users.

The data model of the I2NSF Consumer-Facing Interface utilizes the NACM's mechanisms to manage the access control on the I2NSF Consumer-Facing Interface. The NACM with the above features can be used to set up the access control rules of a user group in the I2NSF Consumer-Facing Interface.

Figure 17 shows part of the NACM module to enable the access control of a user group for the I2NSF Consumer-Facing Interface. To use the NACM, a user needs to configure either a NETCONF server [RFC6241] or a RESTCONF server [RFC8040] to enable the NACM module. Then, the user can simply use an account of root or admin user for the access control for the module of the I2NSF Consumer-Facing Interface (i.e., ietf-i2nsf-cfi-policy). An XML example to configure the access control a user group for the I2NSF Consumer-Facing Interface can be seen in Section 9.

```
list rule {
  key "name";
  ordered-by user;
  leaf name {
    type string {
      length "1..max";
    }
    description
      "Arbitrary name assigned to the rule.";
  }

  leaf module-name {
    type union {
      type matchall-string-type;
      type string;
    }
    default "*";
    description
      "Name of the module associated with this rule."
  }

  leaf access-operations {
    type union {
      type matchall-string-type;
      type access-operations-type;
    }
    default "*";
    description
      "Access operations associated with this rule."
  }

  leaf action {
    type action-type;
    mandatory true;
    description
      "The access control action associated with the
     rule.  If a rule is determined to match a
     particular request, then this object is used
     to determine whether to permit or deny the
     request.";
  }
```

Figure 17: A Part of the NACM YANG Data Model

7. YANG Data Model of Consumer-Facing Interface

The main objective of this document is to provide both an information model and the corresponding YANG data model of I2NSF

Consumer-Facing Interface. This interface can be used to deliver control and management messages between an I2NSF User and Security Controller for the I2NSF User's high-level security policies.

The semantics of the data model must be aligned with the information model of the Consumer-Facing Interface. The transformation of the information model is performed so that this YANG data model can facilitate the efficient delivery of the control or management messages.

This data model is designed to support the I2NSF framework that can be extended according to the security needs. In other words, the model design is independent of the content and meaning of specific policies as well as the implementation approach.

With the YANG data model of I2NSF Consumer-Facing Interface, this document suggests use cases for security policy rules such as time-based firewall, VoIP/VoLTE security service, and DDoS-attack mitigation in Section 8.

7.1. YANG Module of Consumer-Facing Interface

This section describes a YANG module of Consumer-Facing Interface. This document provides identities in the data model to be used for configuration of an NSF. Each identity is used for a different type of configuration. The details are explained in the description of each identity. This YANG module imports from [RFC6991]. It makes references to [RFC0768][RFC0792][RFC0793] [RFC0854][RFC0959] [RFC1939] [RFC2818][RFC3022][RFC3261] [RFC3501][RFC4250][RFC4340] [RFC4443][RFC5321][RFC7230] [RFC7231][I-D.ietf-i2nsf-capability] [I-D.ietf-tcpm-rfc793bis][IANA-ICMP-Parameters] [IANA-ICMPv6-Parameters][Encyclopedia-Britannica] [STIX].

```
<CODE BEGINS> file "ietf-i2nsf-cfi-policy@2021-09-15.yang"

module ietf-i2nsf-cfi-policy {
  yang-version 1.1;
  namespace
    "urn:ietf:params:xml:ns:yang:ietf-i2nsf-cfi-policy";
  prefix nsfcfi;

  import ietf-inet-types{
    prefix inet;
    reference "RFC 6991";
  }

  import ietf-yang-types{
    prefix yang;
    reference "RFC 6991";
  }

  organization
    "IETF I2NSF (Interface to Network Security Functions)
     Working Group";

  contact
    "WG Web: <https://tools.ietf.org/wg/i2nsf>
     WG List: <mailto:i2nsf@ietf.org>

     Editor: Jaehoon Paul Jeong
     <mailto:pauljeong@skku.edu>

     Editor: Patrick Lingga
     <mailto:patricklink@skku.edu>";

  description
    "This module is a YANG module for Consumer-Facing Interface.

     Copyright (c) 2021 IETF Trust and the persons identified as
     authors of the code. All rights reserved.

     Redistribution and use in source and binary forms, with or
     without modification, is permitted pursuant to, and subject to
     the license terms contained in, the Simplified BSD License set
     forth in Section 4.c of the IETF Trust's Legal Provisions
     Relating to IETF Documents
     (https://trustee.ietf.org/license-info).

     This version of this YANG module is part of RFC XXXX
     (https://www.rfc-editor.org/info/rfcXXXX); see the RFC itself
     for full legal notices.";

  // RFC Ed.: replace XXXX with an actual RFC number and remove
  // this note.

  revision "2021-09-15" {
    description "Initial revision.";
    reference
      "RFC XXXX: I2NSF Consumer-Facing Interface YANG Data Model";

    // RFC Ed.: replace XXXX with an actual RFC number and remove
    // this note.
  }

  identity resolution-strategy {
    description
      "Base identity for resolution strategy";
    reference
      "draft-ietf-i2nsf-capability-data-model-17:
       I2NSF Capability YANG Data Model - Resolution Strategy";
  }

  identity fmr {
    base resolution-strategy;
    description
      "Identity for First Matching Rule (FMR)";
    reference
      "draft-ietf-i2nsf-capability-data-model-17:
       I2NSF Capability YANG Data Model - Resolution Strategy";
  }

  identity lmr {
    base resolution-strategy;
    description
      "Identity for Last Matching Rule (LMR)";
    reference
      "draft-ietf-i2nsf-capability-data-model-17:
       I2NSF Capability YANG Data Model - Resolution Strategy";
  }

  identity pmr {
    base resolution-strategy;
    description
      "Identity for Prioritized Matching Rule (PMR)";
    reference
      "draft-ietf-i2nsf-capability-data-model-17:
       I2NSF Capability YANG Data Model - Resolution Strategy";
  }

  identity pmre {
    base resolution-strategy;
    description
```

```
    "Identity for Prioritized Matching Rule
     with Errors (PMRE)";
  reference
    "draft-ietf-i2nsf-capability-data-model-17:
     I2NSF Capability YANG Data Model - Resolution Strategy";
}

identity pmrn {
  base resolution-strategy;
  description
    "Identity for Prioritized Matching Rule
     with No Errors (PMRN)";
  reference
    "draft-ietf-i2nsf-capability-data-model-17:
     I2NSF Capability YANG Data Model - Resolution Strategy";
}

identity security-event {
  description
    "Base identity for security event types.";
}

identity anti-ddos {
  base security-event;
  description
    "Identity for Anti-DDoS event types.";
}

identity ips {
  base security-event;
  description
    "Identity for Intrusion Prevention System event types.";
}

identity url-filtering {
  base security-event;
  description
    "Identity for url-filtering event types.";
}

identity anti-virus {
  base security-event;
  description
    "Identity for Antivirus types.";
}

identity voip-volte-filtering {
  base security-event;
  description
```

```
    "Identity for VoIP/VoLTE Filtering event types.";
}

identity protocol {
  description
    "This identity represents the protocol types.";
}

identity transport-protocol {
  base protocol;
  description
    "Base identity for the Layer 4 (i.e., Transport Layer)
    Protocols";
}

identity tcp {
  base transport-protocol;
  description
    "Base identity for TCP condition capabilities";
  reference
    "RFC 793: Transmission Control Protocol
     draft-ietf-tcpm-rfc793bis: Transmission Control Protocol
     (TCP) Specification";
}

identity udp {
  base transport-protocol;
  description
    "Base identity for UDP condition capabilities";
  reference
    "RFC 768: User Datagram Protocol";
}

identity sctp {
  base transport-protocol;
  description
    "Identity for SCTP condition capabilities";
  reference
    "RFC 4960: Stream Control Transmission Protocol";
}

identity dccp {
  base transport-protocol;
  description
    "Identity for DCCP condition capabilities";
  reference
    "RFC 4340: Datagram Congestion Control Protocol";
}
```

```
identity application-protocol {
  base protocol;
  description
    "Base identity for the Layer 7 (i.e., Application Layer)
     Protocols";
}

identity ftp {
  base application-protocol;
  description
    "The identity for ftp protocol.";
  reference
    "RFC 959: File Transfer Protocol (FTP)";
}

identity ssh {
  base application-protocol;
  description
    "The identity for ssh protocol.";
  reference
    "RFC 4250: The Secure Shell (SSH) Protocol";
}

identity telnet {
  base application-protocol;
  description
    "The identity for telnet.";
  reference
    "RFC 854: Telnet Protocol";
}

identity smtp {
  base application-protocol;
  description
    "The identity for smtp.";
  reference
    "RFC 5321: Simple Mail Transfer Protocol (SMTP)";
}

identity http {
  base application-protocol;
  description
    "The identity for http.";
  reference
    "RFC7230: Hypertext Transfer Protocol (HTTP/1.1): Message
     Syntax and Routing
     RFC7231: Hypertext Transfer Protocol (HTTP/1.1): Semantics
     and Content";
}
```

```
identity https {
  base application-protocol;
  description
    "The identity for https.";
  reference
    "RFC 2818: HTTP over TLS (HTTPS)";
}

identity pop3 {
  base application-protocol;
  description
    "The identity for pop3.";
  reference
    "RFC 1939: Post Office Protocol - Version 3 (POP3)";
}

identity imap {
  base application-protocol;
  description
    "The identity for Internet Message Access Protocol (IMAP).";
  reference
    "RFC 3501: INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1";
}

identity action {
  description
    "Base identity for action";
}

identity ingress-action {
  base action;
  description
    "Base identity to represents the ingress actions, such as
     pass, drop, rate-limit, and mirror.";
}

identity egress-action {
  base action;
  description
    "Base identity represents the egress actions, such as
     pass, drop, rate-limit, mirror, invoke-signaling,
     tunnel-encapsulation, forwarding, and transformation.";
}

identity pass {
  base ingress-action;
  description
    "The identity for pass.";
}

identity drop {
  base ingress-action;
  description
    "The identity for drop.";
}

identity rate-limit {
  base ingress-action;
  description
    "The identity for rate-limit.";
}

identity mirror {
  base ingress-action;
  description
    "The identity for mirroring.";
}

identity invoke-signaling {
  base egress-action;
  description
    "Identity for invoke signaling action capability";
  reference
    "RFC 8329: Framework for Interface to Network Security
     Functions - Invoke-signaling action";
}

identity tunnel-encapsulation {
  base egress-action;
  description
    "Identity for tunnel encapsulation action capability";
  reference
    "RFC 8329: Framework for Interface to Network Security
     Functions - Tunnel Encapsulation";
}

identity forwarding {
  base egress-action;
  description
    "Identity for forwarding action capability";
  reference
    "RFC 8329: Framework for Interface to Network Security
     Functions - Forwarding action";
}

identity transformation {
  base egress-action;
  description
    "Identity for transformation action capability";
  reference
    "RFC 8329: Framework for Interface to Network Security
     Functions - Redirection action";
}

identity log-action {
  description
    "Base identity for representing log actions, such as rule-log
     and session-log action.";
}

identity rule-log {
  base log-action;
  description
    "Identity for rule log-action capability.
     Log the received packet based on the rule";
}

identity session-log {
  base log-action;
  description
    "Identity for session log-action capability.
     Log the received packet based on the session.";
}

identity signature-type {
  description
    "This represents the base identity for signature types.";
}

identity signature-yara {
  base signature-type;
  description
    "This represents the YARA signatures.";
  reference
    "YARA: YARA signatures are explained.";
}

identity signature-snort {
  base signature-type;
  description
    "This represents the SNORT signatures.";
  reference
    "SNORT: SNORT signatures are explained.";
}

identity signature-suricata {
```

```
  base signature-type;
  description
    "This represents the SURICATA signatures.";
  reference
    "SURICATA: SURICATA signatures are explained.";
}

identity threat-feed-type {
  description
    "This represents the base identity for threat-feed.";
}

identity day {
  description
    "This represents the base for days.";
}

identity monday {
  base day;
  description
    "This represents Monday.";
}

identity tuesday {
  base day;
  description
    "This represents Tuesday.";
}

identity wednesday {
  base day;
  description
    "This represents Wednesday.";
}

identity thursday {
  base day;
  description
    "This represents Thursday.";
}

identity friday {
  base day;
  description
    "This represents Friday.";
}

identity saturday {
  base day;
  description
```

```
    "This represents Saturday.";
}

identity sunday {
  base day;
  description
    "This represents Sunday.";
}

identity continent {
  description
    "Base identity for continent types. The continents are based
     on Encyclopedia Britannica";
  reference
    "Encyclopedia Britannica: Continent";
}

identity africa {
  base continent;
  description
    "Identity for Africa.";
  reference
    "Encyclopedia Britannica: Continent";
}

identity asia {
  base continent;
  description
    "Identity for Asia.";
  reference
    "Encyclopedia Britannica: Continent";
}

identity antarctica {
  base continent;
  description
    "Identity for Antarctica.";
  reference
    "Encyclopedia Britannica: Continent";
}

identity europe {
  base continent;
  description
    "Identity for Europe.";
  reference
    "Encyclopedia Britannica: Continent";
}
```

```
 identity north-america {
   base continent;
   description
     "Identity for North America.";
   reference
     "Encyclopedia Britannica: Continent";
 }

 identity south-america {
   base continent;
   description
     "Identity for South America.";
   reference
     "Encyclopedia Britannica: Continent";
 }

 identity australia {
   base continent;
   description
     "Identity for Australia";
   reference
     "Encyclopedia Britannica: Continent";
 }

/*
 * Typedefs
 */
 typedef time {
   type string {
     pattern '(0[0-9]|1[0-9]|2[0-3]):[0-5][0-9]:[0-5][0-9](\.\d+)?'
       + '(Z|[\+\-]((1[0-3]|0[0-9]):([0-5][0-9])|14:00))?';
   }
   description
     "The time type represents an instance of time of zero-duration
      that recurs every day.";
 }

/*
 * Groupings
 */

 grouping ipv4-list {
   description
     "Grouping for an IPv4 address list.";
   leaf-list ipv4 {
     type inet:ipv4-address-no-zone;
     description
       "This is the entry for an IPv4 address list.";
   }
```

```
    }

    grouping ipv6-list {
      description
        "Grouping for an IPv6 address list.";
      leaf-list ipv6 {
        type inet:ipv6-address-no-zone;
        description
          "This is the entry for an IPv6 address list.";
      }
    }

    grouping ipv4 {
      description
        "Grouping for an IPv4 address.";
      leaf ipv4 {
        type inet:ipv4-address-no-zone;
        description
          "This is the entry for an IPv4 address.";
      }
    }

    grouping ipv6 {
      description
        "Grouping for an IPv6 address.";
      leaf ipv6 {
        type inet:ipv6-address-no-zone;
        description
          "This is the entry for an IPv6 address.";
      }
    }

    grouping ip-address-info {
      description
        "There are two types to configure a security policy
         for an IP address, such as IPv4 adress and IPv6 address.";
      choice match-type {
        description
          "User can choose between IPv4 and IPv6.";
        case range-match-ipv4 {
          container range-ipv4-address {
            leaf start-ipv4-address {
              type inet:ipv4-address-no-zone;
              mandatory true;
              description
                "A start IPv4 address for a range match.";
            }
            leaf end-ipv4-address {
              type inet:ipv4-address-no-zone;
              mandatory true;
              description
                "An end IPv4 address for a range match.";
            }
            description
              "A range match for IPv4 addresses is provided.
               Note that the start IPv4 address must be lower than
               the end IPv4 address.";
          }
        }
        case range-match-ipv6 {
          container range-ipv6-address {
            leaf start-ipv6-address {
              type inet:ipv6-address-no-zone;
              mandatory true;
              description
                "A start IPv6 address for a range match.";
            }
            leaf end-ipv6-address {
              type inet:ipv6-address-no-zone;
              mandatory true;
              description
                "An end IPv6 address for a range match.";
            }
            description
              "A range match for IPv6 addresses is provided.
               Note that the start IPv6 address must be lower than
               the end IPv6 address.";
          }
        }
      }
    }

    grouping user-group {
      description
        "This group represents user group information such as name and
         ip-address.";
      leaf name {
        type string;
        description
          "This represents the name of a user-group.  A user-group name
           is used to map a user-group's name (e.g., employees) to IP
           address(es), MAC address(es).
           It is dependent on implementation.";
      }
      leaf-list mac-address {
        type yang:mac-address;
        description
          "Represent the MAC Address of a user-group. A user-group
```

```
       can have multiple MAC Addresses.";
  }
  uses ip-address-info{
    description
      "This represents the IP addresses of a user-group.";
    refine match-type{
      mandatory true;
    }
  }
}

grouping device-group {
  description
    "This group represents device group information such as
     ip-address protocol.";
  leaf name {
    type string;
    description
      "This represents the name of a device-group.";
  }
  uses ip-address-info{
    refine match-type{
      mandatory true;
    }
  }
  leaf-list application-protocol {
    type identityref {
      base application-protocol;
    }
    description
      "This represents the application layer protocols of devices.
       If this is not set, it cannot support the appropriate
       protocol";
  }
}

grouping location-group {
  description
    "This group represents location-group information such as
     geo-ip and continent.";
  leaf name {
    type string;
    description
      "This represents the name of a location.";
  }
  list geo-ip-ipv4 {
    key "ipv4-address";
    description
      "This represents the list of IPv4 addresses based on a
```

```
       location.";
    leaf ipv4-address{
      type inet:ipv4-address-no-zone;
      description
        "This represents an IPv4 geo-ip address of a location.";
    }
    leaf ipv4-prefix{
      type inet:ipv4-prefix;
      description
        "This represents the prefix for the IPv4 addresses.";
    }
  }
  list geo-ip-ipv6 {
    key "ipv6-address";
    description
      "This represents the list of IPv6 addresses based on a
       location.";
    leaf ipv6-address{
      type inet:ipv6-address-no-zone;
      description
        "This represents an IPv6 geo-ip address of a location.";
    }
    leaf ipv6-prefix{
      type inet:ipv6-prefix;
      description
        "This represents the prefix for the IPv6 addresses.";
    }
  }
  leaf continent {
    type identityref {
      base continent;
    }
    default asia;
    description
      "location-group has geo-ip addresses of the corresponding
       continent.";
  }
}

grouping payload-string {
  description
    "The grouping for payload-string content.  It contains
     information such as name and string content.";
  leaf description {
    type string;
    description
      "This represents the description of a payload.  If this is
       not set, it cannot support the description of how the
       payload content is related to a security attack.";
  }
  leaf-list content {
    type string;
    description
      "This represents the string of the payload contents.
       This content leaf-list contains the payload of a packet to
       analyze a threat. Due to the types of threats, the type of
       the content is defined as a string to accommodate any kind
       of a payload type such as HTTP, HTTPS, and SIP.  If this is
       not set, it cannot support the payload contents involved in
       a security attack as a string.";
  }
}

list i2nsf-cfi-policy {
  key "policy-name";
  description
    "This is a security policy list.  Each policy in the list
     contains a list of security policy rules, and is a policy
     instance to have the information of where and when a policy
     needs to be applied.";
  leaf policy-name {
    type string;
    description
      "The name which identifies the policy.";
  }
  leaf resolution-strategy {
    type identityref {
      base resolution-strategy;
    }
    default fmr;
    description
      "The resolution strategies that can be used to
      specify how to resolve conflicts that occur between
      actions of the same or different policy rules that
      are matched and contained in this particular NSF";

    reference
      "draft-ietf-i2nsf-capability-data-model-17:
       I2NSF Capability YANG Data Model - Resolution strategy";
  }
  list rules {
    key "rule-name";

    description
      "There can be a single or multiple number of rules.";
    leaf rule-name {
      type string;
      description
    "This represents the name for a rule.";
}

leaf priority {
  type uint8 {
    range "1..255";
  }
  description
    "The priority keyword comes with a mandatory
     numeric value which can range from 1 through 255.
     Note that a higher number means a higher priority";
}

container event {
  description
    "This represents an event (i.e., a security event), for
     which a security rule is made.";
  leaf security-event {
    type identityref {
      base security-event;
    }
    description
      "This contains the description of a security event.  If
       this is not set, it cannot support what security event
       will be enforced.";
  }

  container time {
    description
      "The time when a security policy rule should be
       applied.";
    leaf start-date-time {
      type yang:date-and-time;
      description
        "This is the start date and time for a security policy
         rule.";
    }
    leaf end-date-time {
      type yang:date-and-time;
      description
        "This is the end date and time for a policy rule.  The
         policy rule will stop working after the specified
         end-date-time.";
    }
    container period {
      when
        "../frequency!='only-once'";
      description
        "This represents the repetition time.  In the case
     where the frequency is weekly, the days can be set.";
  leaf start-time {
    type time;
    description
      "This is a period's start time for an event.";
  }
  leaf end-time {
    type time;
    description
      "This is a period's end time for an event.";
  }
  leaf-list day {
    when
      "../../frequency='weekly'";
    type identityref{
      base day;
    }
    min-elements 1;
    description
      "This represents the repeated day of every week
      (e.g., Monday and Tuesday).  More than one day can be
       specified.";
  }
  leaf-list date {
    when
      "../../frequency='monthly'";
    type int32{
      range "1..31";
    }
    min-elements 1;
    description
      "This represents the repeated date of every month.
       More than one date can be specified.";
  }
  leaf-list month {
    when
      "../../frequency='yearly'";
    type string{
      pattern '\d{2}-\d{2}';
    }
    min-elements 1;
    description
      "This represents the repeated date and month of every
       year.  More than one can be specified.  A pattern
       used here is Month and Date (MM-DD).";
  }
}

leaf frequency {
      type enumeration {
        enum only-once {
          description
            "This represents that the rule is immediately
             enforced only once and not repeated. The policy
             will continuously be active from the start-time
             to the end-time.";
        }
        enum daily {
          description
            "This represents that the rule is enforced on a
             daily basis. The policy will be repeated daily
             until the end-date.";
        }
        enum weekly {
          description
            "This represents that the rule is enforced on a
             weekly basis. The policy will be repeated weekly
             until the end-date. The repeated days can be
             specified.";
        }
        enum monthly {
          description
            "This represents that the rule is enforced on a
             monthly basis. The policy will be repeated monthly
             until the end-date.";
        }
        enum yearly {
          description
            "This represents that the rule is enforced on a
             yearly basis.  The policy will be repeated yearly
             until the end-date.";
        }
      }
      default only-once;
      description
        "This represents how frequently the rule should be
         enforced.";
    }
  }
}

container condition {
  description
  "Conditions for general security policies.";
  container firewall-condition {
    description
      "A general firewall condition based on the packet
       header.";
```

```
leaf-list source {
  type union {
    type leafref {
      path
        "/i2nsf-cfi-policy/endpoint-groups/"
       +"user-group/name";
    }
    type leafref {
      path
        "/i2nsf-cfi-policy/endpoint-groups/"
       +"device-group/name";
    }
  }
  description
  "This describes the path of the source.";
}

leaf-list destination {
  type union {
    type leafref {
      path
        "/i2nsf-cfi-policy/endpoint-groups/"
       +"user-group/name";
    }
    type leafref {
      path
        "/i2nsf-cfi-policy/endpoint-groups/"
       +"device-group/name";
    }
  }
  description
    "This describes the path to the destinations.";
}

leaf transport-layer-protocol {
  type identityref {
    base transport-protocol;
  }
  description
    "The transport-layer protocol to be matched.";
}

container range-port-number {
  leaf start-port-number {
    type inet:port-number;
    description
      "A start port number for range match.";
  }
  leaf end-port-number {
    type inet:port-number;
    description
      "An end port number for range match.";
  }
  description
    "A range match for transport-layer port number. Note
     that the start port number value must be lower than
     the end port  number value";
}

list icmp {
  key "version";
  description
     "Represents the ICMP packet header information to
      determine if the set of policy actions in this ECA
      policy rule should be executed or not.";
   reference
     "RFC  792: Internet Control Message Protocol
      RFC 8335: PROBE: A Utility for Probing Interfaces";

  leaf version {
    type enumeration {
      enum icmpv4 {
        value "1";
        description
          "The ICMPv4 Protocol as defined in RFC 792";
      }
      enum icmpv6 {
        value "2";
        description
          "The ICMPv6 Protocol as defined in RFC 4443";
      }
    }
    description
      "The ICMP version to be matched. This value
       affected the type and code values.";
    reference
      "RFC  792: Internet Control Message Protocol
       RFC 4443: Internet Control Message Protocol (ICMPv6)
       for the Internet Protocol Version 6 (IPv6)
       Specification";
  }

  leaf-list type {
    type uint8;
    description
      "The security policy rule according to
       ICMP type parameter.";
    reference
```

```
        "RFC  792: Internet Control Message Protocol
         RFC 8335: PROBE: A Utility for Probing Interfaces
         IANA: Internet Control Message Protocol (ICMP)
         Parameters
         IANA: Internet Control Message Protocol version 6
         (ICMPv6) Parameters";
    }

    leaf-list code {
      type uint8;
      description
        "The security policy rule according to
         ICMP code parameter.";
      reference
        "RFC  792: Internet Control Message Protocol
         RFC 8335: PROBE: A Utility for Probing Interfaces
         IANA: Internet Control Message Protocol (ICMP)
         Parameters
         IANA: Internet Control Message Protocol version 6
         (ICMPv6) Parameters";
    }
  }
}

container ddos-condition {
  description
    "A condition for a DDoS attack.";
  container rate-limit {
    description
      "This describes the rate-limit.";
    leaf packet-rate-threshold {
      type uint32;

      description
        "This is a trigger value for a rate limit of packet
         rate for a DDoS-attack mitigation.";
    }
    leaf byte-rate-threshold {
      type uint32;
      description
        "This is a trigger value for a rate limit of byte
         rate for a  DDoS-attack mitigation.";
    }
    leaf flow-rate-threshold {
      type uint32;
      description
        "This is a trigger value for a rate limit of flow
         rate for a DDoS-attack mitigation.";
    }
  }
}

container anti-virus-condition {
  description
   "A condition for anti-virus";

  leaf-list exception-files {
    type string;
    description
      "The type or name of the files to be excluded by the
       anti-virus. This can be used to keep the known
       harmless files.";
  }
}

container payload-condition {
  description
    "A condition based on a packet's content.";
  leaf-list content {
    type leafref {
      path "/i2nsf-cfi-policy/threat-preventions/"
         + "payload-content/name";
    }
    description
      "This describes the paths to a packet content's";
  }
}

container url-condition {
  description
    "Condition for url category";
  leaf url-name {
    type leafref {
      path
        "/i2nsf-cfi-policy/endpoint-groups/"
       +"url-group/name";
    }
    description
      "This is description for the condition of a URL's
       category such as SNS sites, game sites, ecommerce
       sites, company sites, and university sites.";
  }
}

container voice-condition {
  description
    "For the VoIP/VoLTE security system, a 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.";
  reference
    "RFC 3261: SIP: Session Initiation Protocol";

  leaf-list source-id {
    type string;
    description
      "The security policy rule according to
       a source voice ID for VoIP and VoLTE.";
  }

  leaf-list destination-id {
    type string;
    description
      "The security policy rule according to
       a destination voice ID for VoIP and VoLTE.";
  }

  leaf-list user-agent {
    type string;
    description
      "The security policy rule according to
       an user agent for VoIP and VoLTE.";
  }
}

container context-condition {
  description
    "Condition for matching the context of the packet, such
     as geographic location, time, packet direction";
  container geography-location-condition {
    description
      "A condition for a location-based connection";
    leaf-list source {
      type leafref {
        path
          "/i2nsf-cfi-policy/endpoint-groups/"
         +"location-group/name";
      }
      description
        "This describes the paths to a location's sources.";
    }
```

```
      leaf-list destination {
        type leafref {
          path
            "/i2nsf-cfi-policy/endpoint-groups/"
           +"location-group/name";
        }
        description
          "This describes the paths to a location's
           destinations.";
      }
    }
  }

  container threat-feed-condition {
    description
      "A condition based on the threat-feed information.";
    leaf-list name {
      type leafref {
        path
          "/i2nsf-cfi-policy/threat-preventions/"
         +"threat-feed-list/name";
      }
      description
        "This describes the paths to a threat-feed's sources.";
    }
  }
}

container actions {
  description
    "This is the action container.";
  container primary-action {
    description
      "This represent primary actions (e.g., ingress and egress
       action) to be applied to a condition.
       If this is not set, it cannot support the primary
       actions.";
    leaf action {
      type identityref {
        base action;
      }
      description
        "Ingress Action: pass, drop, reject, rate-limit,
         and mirror.
         Egress action: mirror, invoke-signaling,
         tunnel-encapsulation, forwarding, and redirection.";
    }
  }
  container secondary-action {
      description
        "This represents secondary actions (e.g., log and syslog)
         to be applied if they are needed.  If this is not set,
         it cannot support the secondary actions.";
      leaf log-action {
        type identityref {
          base log-action;
        }
        description
          "Log action: rule log and session log";
      }
    }
  }
}

container endpoint-groups {
  description
    "A logical entity in a business environment, where a security
     policy is to be applied.";
  list user-group{
    uses user-group;
    key "name";
    description
      "This represents a user group.";
  }
  list device-group {
    key "name";
    uses device-group;
    description
      "This represents a device group.";
  }
  list location-group{
    key "name";
    uses location-group;
    description
      "This represents a location group.";
  }
  list url-group {
    key "name";
    description
      "This describes the list of URL.";
    leaf name {
      type string;
      description
        "This is the name of URL group, e.g., SNS sites,
         gaming sites, ecommerce sites";
    }
    leaf-list url {
      type string;
      description
        "Specifies the URL to be added into the group.";
    }
  }
}

container threat-preventions {
  description
    "This describes the list of threat-preventions.";
  list threat-feed-list {
    key "name";
    description
      "There can be a single or multiple number of
       threat-feeds.";
    leaf name {
      type string;
      description
        "This represents the name of the threat-feed.";
    }
    leaf description {
      type string;
      description
        "This represents the descriptions of a threat-feed.  The
         description should include information, such as type,
         threat, method, and file type.  Structured Threat
         Information Expression (STIX) can be used for
         description of a threat [STIX].";
    }
    leaf-list signatures {
      type identityref {
        base signature-type;
      }
      description
        "This contains a list of signatures or hashes of the
         threats.";
    }
  }
  list payload-content {
    key "name";
    leaf name {
      type string;
      description
        "This represents the name of a packet's payload-content.
         It should give an idea of why a specific payload content
         is marked as a threat.  For example, the name 'backdoor'
         indicates the payload content is related to a backdoor
         attack.";
    }
    description
```

```
          "This represents a payload-string group.";
        uses payload-string;
      }
    }
  }
}

<CODE ENDS>
```

Figure 18: YANG for Consumer-Facing Interface

8. XML Configuration Examples of High-Level Security Policy Rules

This section shows XML configuration examples of high-level security policy rules that are delivered from the I2NSF User to the Security Controller over the Consumer-Facing Interface. The considered use cases are: Database registration, time-based firewall for web filtering, VoIP/VoLTE security service, and DDoS-attack mitigation.

8.1. Database Registration: Information of Positions and Devices (Endpoint Group)

If new endpoints are introduced to the network, it is necessary to first register their data to the database. For example, if new members are newly introduced in either of three different groups (i.e., user-group, device-group, and url-group), each of them should be registered with information such as ip-addresses or protocols used by devices.

Figure 19 shows an example XML representation of the registered information for the user-group and device-group with IPv4 addresses [RFC5737].

```
<?xml version="1.0" encoding="UTF-8" ?>
<i2nsf-cfi-policy
 xmlns="urn:ietf:params:xml:ns:yang:ietf-i2nsf-cfi-policy">
  <endpoint-groups>
    <user-group>
      <name>employees</name>
      <range-ipv4-address>
        <start-ipv4-address>192.0.2.11</start-ipv4-address>
        <end-ipv4-address>192.0.2.90</end-ipv4-address>
      </range-ipv4-address>
    </user-group>
    <device-group>
      <name>webservers</name>
      <range-ipv4-address>
        <start-ipv4-address>198.51.100.11</start-ipv4-address>
        <end-ipv4-address>198.51.100.20</end-ipv4-address>
      </range-ipv4-address>
      <protocol>nsfcfi:http</protocol>
      <protocol>nsfcfi:https</protocol>
    </device-group>
    <url-group>
      <name>sns-websites</name>
      <user-defined>SNS_1</user-defined>
      <user-defined>SNS_2</user-defined>
    </url-group>
  </endpoint-groups>
</i2nsf-cfi-policy>
```

Figure 19: Registering User-group and Device-group Information with IPv4 Addresses

Also, Figure 20 shows an example XML representation of the registered information for the user-group and device-group with IPv6 addresses [RFC3849].

```
<?xml version="1.0" encoding="UTF-8" ?>
<i2nsf-cfi-policy
 xmlns="urn:ietf:params:xml:ns:yang:ietf-i2nsf-cfi-policy">
  <endpoint-groups>
    <user-group>
      <name>employees</name>
      <range-ipv6-address>
        <start-ipv6-address>2001:DB8:0:1::11</start-ipv6-address>
        <end-ipv6-address>2001:DB8:0:1::90</end-ipv6-address>
      </range-ipv6-address>
    </user-group>
    <device-group>
      <name>webservers</name>
      <range-ipv6-address>
        <start-ipv6-address>2001:DB8:0:2::11</start-ipv6-address>
        <end-ipv6-address>2001:DB8:0:2::20</end-ipv6-address>
      </range-ipv6-address>
      <protocol>nsfcfi:http</protocol>
      <protocol>nsfcfi:https</protocol>
    </device-group>
    <url-group>
      <name>sns-websites</name>
      <url>SNS_1</url>
      <url>SNS_2</url>
    </url-group>
  </endpoint-groups>
</i2nsf-cfi-policy>
```

Figure 20: Registering User-group and Device-group Information with IPv6 Addresses

8.2. Scenario 1: Block SNS Access during Business Hours

The first example scenario is to "block SNS access during office hours" using a time-based firewall policy. In this scenario, all users registered as "employees" in the user-group list are unable to access Social Networking Services (SNS) during the office hours (weekdays). The XML instance is described below:

```
<?xml version="1.0" encoding="UTF-8" ?>
<i2nsf-cfi-policy
 xmlns="urn:ietf:params:xml:ns:yang:ietf-i2nsf-cfi-policy">
  <policy-name>security_policy_for_blocking_sns123</policy-name>
  <rules>
    <rule-name>block_access_to_sns_during_office_hours</rule-name>
    <event>
      <time-information>
        <start-date-time>2021-03-11T09:00:00.00Z</start-date-time>
        <end-date-time>2021-12-31T18:00:00.00Z</end-date-time>
        <period>
          <start-time>09:00:00Z</start-time>
          <end-time>18:00:00Z</end-time>
          <day>nsfcfi:monday</day>
          <day>nsfcfi:tuesday</day>
          <day>nsfcfi:wednesday</day>
          <day>nsfcfi:thursday</day>
          <day>nsfcfi:friday</day>
        </period>
      </time-information>
      <frequency>weekly</frequency>
    </event>
    <condition>
      <firewall-condition>
        <source>employees</source>
      </firewall-condition>
      <url-condition>
        <url-name>sns-websites</url-name>
      </url-condition>
    </condition>
    <actions>
      <primary-action>nsfcfi:drop</primary-action>
    </actions>
  </rules>
</i2nsf-cfi-policy>
```

Figure 21: An XML Example for Time-based Firewall

Time-based-condition Firewall

1. The policy name is "security_policy_for_blocking_sns".

2. The rule name is "block_access_to_sns_during_office_hours".

3. The Source is "employees".

4. The destination target is "sns-websites". "sns-websites" is the key which represents the list containing the information, such as URL, about sns-websites.

5. The action required is to "drop" any attempt to connect to websites related to Social networking.

8.3. Scenario 2: Block Malicious VoIP/VoLTE Packets Coming to a Company

The second example scenario is to "block malicious VoIP/VoLTE packets coming to a company" using a VoIP policy. In this scenario, the calls comming from from VOIP and/or VOLTE sources with VOLTE IDs that are classified as malicious are dropped. The IP addresses of the employees and malicious VOIP IDs should be blocked are stored in the database or datastore of the enterprise. Here and the rest of the cases assume that the security administrators or someone responsible for the existing and newly generated policies, are not aware of which and/or how many NSFs are needed to meet the security requirements. Figure 22 represents the XML document generated from YANG discussed in previous sections. Once a high-level seucurity policy is created by a security admin, it is delivered by the Consumer-Facing Interface, through RESTCONF server, to the security controller. The XML instance is described below:

```
<?xml version="1.0" encoding="UTF-8" ?>
<i2nsf-cfi-policy
 xmlns="urn:ietf:params:xml:ns:yang:ietf-i2nsf-cfi-policy">
  <policy-name>
    security_policy_for_blocking_malicious_voip_packets
  </policy-name>
  <rules>
    <rule-name>Block_malicious_voip_and_volte_packets</rule-name>
    <condition>
      <voice-condition>
        <source-id>malicious-id</source-id>
      </voice-condition>
      <firewall-condition>
        <destination>employees</destination>
      </firewall-condition>
    </condition>
    <actions>
      <primary-action>nsfcfi:drop</primary-action>
    </actions>
  </rules>
</i2nsf-cfi-policy>
```

Figure 22: An XML Example for VoIP Security Service

Custom-condition Firewall

1. The policy name is "security_policy_for_blocking_malicious_voip_packets".

2. The rule name is "Block_malicious_voip_and_volte_packets".

3. The Source is "malicious-id". This can be a single ID or a list of IDs, depending on how the ID are stored in the database. The "malicious-id" is the key so that the security admin can read every stored malicious VOIP IDs that are named as "malicious-id".

4. The destination target is "employees". "employees" is the key which represents the list containing information about employees, such as IP addresses.

5. The action required is "drop" when any incoming packets are from "malicious-id".

8.4. Scenario 3: Mitigate HTTP and HTTPS Flood Attacks on a Company Web Server

The third example scenario is to "Mitigate HTTP and HTTPS flood attacks on a company web server" using a DDoS-attack mitigation policy. Here, the time information is not set because the service provided by the network should be maintained at all times. If the packets sent by any sources are more than the set threshold, then the admin can set the percentage of the packets to be dropped to safely maintain the service. In this scenario, the source is set as "any" to block any sources which send abnormal amount of packets. The destination is set as "web_server01". Once the rule is set and delivered and enforced to the nsfs by the securiy controller, the NSFs will monitor the incoming packet amounts and the destination to act according to the rule set. The XML instance is described below:

```
<?xml version="1.0" encoding="UTF-8" ?>
<i2nsf-cfi-policy
  xmlns="urn:ietf:params:xml:ns:yang:ietf-i2nsf-cfi-policy">
  <policy-name>security_policy_for_ddos_attacks</policy-name>
  <rules>
    <rule-name>1000_packets_per_second</rule-name>
    <conditions>
      <ddos-condition>
        <rate-limit>
          <packet-rate-threshold>1000</packet-rate-threshold>
        </rate-limit>
      </ddos-condition>
    </conditions>
    <actions>
      <primary-action>nsfcfi:drop</primary-action>
    </actions>
  </rules>
</i2nsf-cfi-policy>
```

Figure 23: An XML Example for DDoS-attack Mitigation

DDoS-condition Firewall

1. The policy name is "security_policy_for_ddos_attacks".

2. The rule name is "100_packets_per_second".

3. The rate limit exists to limit the incoming amount of packets per second. In this case the rate limit is "1000" packets per second. This amount depends on the packet receiving capacity of the server devices.

4. The Source is all sources which send abnormal amount of packets.

5. The action required is to "drop" packet reception is more than 1000 packets per second.

9. XML Configuration Example of a User Group's Access Control for I2NSF Consumer-Facing Interface

This is an example for creating privileges for a group of users (i.e., a user group) to access and use the I2NSF Consumer-Facing Interface to create security policies via the interface. For the access control of the Consumer-Facing Interface, the NACM module can be used. Figure 24 shows an XML example the access control of a user group (named Example-Group) for I2NSF Consumer-Facing Interface A group called Example-Group can be created and configured with NACM for the Consumer-Facing Interface. For Example-Group, a rule list can created with the name of Example-Group-Rules. Example-Group-Rules has two rules of Example-Group-Rule1 and Example-Group-Rule2 as follows. For Example-Group-Rule1, the privilege of "Read" is allowed to Example-Group for the Consumer-Facing Interface. On the other hand, for Example-Group-Rule2, the privileges of "Create", "Update", and "Delete" are denied against Example-Group for the Consumer-Facing Interface.

```
<?xml version="1.0" encoding="UTF-8" ?>
<nacm xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-acm">
  <enable-nacm>true</enable-nacm>
  <groups>
    <group>
      <name>Example-Group</name>
      <user-name>Alice</user-name>
      <user-name>Bob</user-name>
      <user-name>Eve</user-name>
    </group>
  </groups>
  <rule-list>
    <name>Example-Group-Rules</name>
    <group>Example-Group</group>
    <rule>
      <name>Example-Group-Rule1</name>
      <access-operations>read</access-operations>
      <module-name>ietf-i2nsf-cfi-policy</module-name>
      <action>permit</action>
    </rule>
    <rule>
      <name>Example-Group-Rule2</name>
      <access-operations>create update delete</access-operations>
      <module-name>ietf-i2nsf-cfi-policy</module-name>
      <action>deny</action>
    </rule>
  </rule-list>
</nacm>
```

Figure 24: An XML Example of a User Group's Access Control for I2NSF Consumer-Facing Interface

The access control for the I2NSF Consumer-Facing Interface is as follows.

1. The NACM is enabled.

2. As a group name, Example-Group is specified.

3. As members of the group, Alice, Bob, and Eve are specified.

4. As a rule list name, Example-Group-Rules is specified for managing privileges of Example-Group's members.

5. As the first rule name, Example-Group-Rule1 is specified. This rule is used to give read privilege to Example-Group's members for the module of the I2NSF Consumer-Facing Interface.

6. As the second rule name, Example-Group-Rule2 is specified. This rule is used to deny create, update, and delete privileges against Example-Group's members for the module of the I2NSF Consumer-Facing Interface.

10. IANA Considerations

This document requests IANA to register the following URI in the "IETF XML Registry" [RFC3688]:

```
URI: urn:ietf:params:xml:ns:yang:ietf-i2nsf-cfi-policy
Registrant Contact: The IESG.
XML: N/A; the requested URI is an XML namespace.
```

This document requests IANA to register the following YANG module in the "YANG Module Names" registry [RFC7950][RFC8525]:

```
name: ietf-i2nsf-cfi-policy
namespace: urn:ietf:params:xml:ns:yang:ietf-i2nsf-cfi-policy
prefix: nsfcfi
reference: RFC XXXX

// RFC Ed.: replace XXXX with an actual RFC number and remove
// this note.
```

11. Security Considerations

The YANG module specified in this document defines a data schema designed to be accessed through network management protocols such as NETCONF [RFC6241] or RESTCONF [RFC8040]. The lowest NETCONF layer is the secure transport layer, and the required secure transport is Secure Shell (SSH) [RFC6242]. The lowest RESTCONF layer is HTTPS, and the required secure transport is TLS [RFC8446].

The Network Configuration Access Control Model (NACM) [RFC8341] provides a means of restricting access to specific NETCONF or RESTCONF users to a preconfigured subset of all available NETCONF or RESTCONF protocol operations and contents. Thus, NACM SHOULD be used to restrict the NSF registration from unauthorized users.

There are a number of data nodes defined in this YANG module that are writable, creatable, and deletable (i.e., config true, which is the default). These data nodes may be considered sensitive or vulnerable in some network environments. Write operations to these data nodes could have a negative effect on network and security operations. These data nodes are collected into a single list node with the following sensitivity/vulnerability:

- list i2nsf-cfi-policy: Writing to almost any element of this YANG module would directly impact on the configuration of NSFs, e.g., completely turning off security monitoring and mitigation capabilities; altering the scope of this monitoring and mitigation; creating an overwhelming logging volume to overwhelm downstream analytics or storage capacity; creating logging patterns which are confusing; or rendering useless trained statistics or artificial intelligence models.

Some of the readable data nodes in this YANG module may be considered sensitive or vulnerable in some network environments. It is thus important to control read access (e.g., via get, get-config, or notification) to these data nodes. These are the subtrees and data nodes with their sensitivity/vulnerability:

- list i2nsf-cfi-policy: The leak of this node to an attacker could reveal the specific configuration of security controls to an attacker. An attacker can craft an attack path that avoids observation or mitigations; one may reveal topology information to inform additional targets or enable lateral movement; one enables the construction of an attack path that avoids observation or mitigations; one provides an indication that the operator has discovered the attack. This node also holds a list of endpoint data that is considered private to the users.

12. Acknowledgments

This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea MSIT (Ministry of Science and ICT) (R-20160222-002755, Cloud based Security Intelligence Technology Development for the Customized Security Service Provisioning). This work was supported in part by the IITP (2020-0-00395, Standard Development of Blockchain based Network Management Automation Technology).

13. Contributors

This document is made by the group effort of I2NSF working group. Many people actively contributed to this document, such as Mahdi F. Dachmehchi and Daeyoung Hyun. The authors sincerely appreciate their contributions.

The following are co-authors of this document:

Patrick Lingga Department of Electrical and Computer Engineering Sungkyunkwan University 2066 Seo-ro Jangan-gu Suwon, Gyeonggi-do 16419 Republic of Korea EMail: patricklink@skku.edu

Hyoungshick Kim Department of Computer Science and Engineering Sungkyunkwan University 2066 Seo-ro Jangan-gu Suwon, Gyeonggi-do 16419 Republic of Korea EMail: hyoung@skku.edu

Eunsoo Kim Department of Electronic, Electrical and Computer Engineering Sungkyunkwan University 2066 Seo-ro Jangan-gu Suwon, Gyeonggi-do 16419 Republic of Korea EMail: eskim86@skku.edu

Seungjin Lee Department of Electronic, Electrical and Computer Engineering Sungkyunkwan University 2066 Seo-ro Jangan-gu Suwon, Gyeonggi-do 16419 Republic of Korea EMail: jine33@skku.edu

Jinyong Tim Kim Department of Electronic, Electrical and Computer Engineering Sungkyunkwan University 2066 Seo-ro Jangan-gu Suwon, Gyeonggi-do 16419 Republic of Korea EMail: timkim@skku.edu

Anil Lohiya Juniper Networks 1133 Innovation Way Sunnyvale, CA 94089 US EMail: alohiya@juniper.net

Dave Qi Bloomberg 731 Lexington Avenue New York, NY 10022 US EMail: DQI@bloomberg.net

Nabil Bitar Nokia 755 Ravendale Drive Mountain View, CA 94043 US EMail: nabil.bitar@nokia.com

Senad Palislamovic Nokia 755 Ravendale Drive Mountain View, CA 94043 US EMail: senad.palislamovic@nokia.com

Liang Xia Huawei 101 Software Avenue Nanjing, Jiangsu 210012 China EMail: Frank.Xialiang@huawei.com

14. References

14.1. Normative References

[RFC0768] Postel, J., "User Datagram Protocol", STD 6, RFC 768, DOI 10.17487/RFC0768, August 1980, <https://www.rfc-editor.org/info/rfc768>.

[RFC0792] Postel, J., "Internet Control Message Protocol", STD 5, RFC 792, DOI 10.17487/RFC0792, September 1981, <https://www.rfc-editor.org/info/rfc792>.

[RFC0793] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, DOI 10.17487/RFC0793, September 1981, <https://www.rfc-editor.org/info/rfc793>.

[RFC0854] Postel, J. and J. Reynolds, "Telnet Protocol Specification", STD 8, RFC 854, DOI 10.17487/RFC0854, May 1983, <https://www.rfc-editor.org/info/rfc854>.

[RFC0959] Postel, J. and J. Reynolds, "File Transfer Protocol", STD 9, RFC 959, DOI 10.17487/RFC0959, October 1985, <https://www.rfc-editor.org/info/rfc959>.

[RFC1939] Myers, J. and M. Rose, "Post Office Protocol - Version 3", STD 53, RFC 1939, DOI 10.17487/RFC1939, May 1996, <https://www.rfc-editor.org/info/rfc1939>.

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

[RFC3261]

Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, DOI 10.17487/RFC3261, June 2002, <https://www.rfc-editor.org/info/rfc3261>.

[RFC3501] Crispin, M., "INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1", RFC 3501, DOI 10.17487/RFC3501, March 2003, <https://www.rfc-editor.org/info/rfc3501>.

[RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <https://www.rfc-editor.org/info/rfc3688>.

[RFC4250] Lehtinen, S. and C. Lonvick, Ed., "The Secure Shell (SSH) Protocol Assigned Numbers", RFC 4250, DOI 10.17487/

RFC4250, January 2006, <https://www.rfc-editor.org/info/rfc4250>.

[RFC4340] Kohler, E., Handley, M., and S. Floyd, "Datagram Congestion Control Protocol (DCCP)", RFC 4340, DOI 10.17487/RFC4340, March 2006, <https://www.rfc-editor.org/info/rfc4340>.

[RFC4443] Conta, A., Deering, S., and M. Gupta, Ed., "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", STD 89, RFC 4443, DOI 10.17487/RFC4443, March 2006, <https://www.rfc-editor.org/info/rfc4443>.

[RFC5321] Klensin, J., "Simple Mail Transfer Protocol", RFC 5321, DOI 10.17487/RFC5321, October 2008, <https://www.rfc-editor.org/info/rfc5321>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", RFC 6242, DOI 10.17487/RFC6242, June 2011, <https://www.rfc-editor.org/info/rfc6242>.

[RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 6991, DOI 10.17487/RFC6991, July 2013, <https://www.rfc-editor.org/info/rfc6991>.

[RFC7230] Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing", RFC 7230, DOI 10.17487/RFC7230, June 2014, <https://www.rfc-editor.org/info/rfc7230>.

[RFC7231] Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content", RFC

7231, DOI 10.17487/RFC7231, June 2014, <https://www.rfc-editor.org/info/rfc7231>.

[RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <https://www.rfc-editor.org/info/rfc7950>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8340] Bjorklund, M. and L. Berger, Ed., "YANG Tree Diagrams", BCP 215, RFC 8340, DOI 10.17487/RFC8340, March 2018, <https://www.rfc-editor.org/info/rfc8340>.

[RFC8341] Bierman, A. and M. Bjorklund, "Network Configuration Access Control Model", STD 91, RFC 8341, DOI 10.17487/RFC8341, March 2018, <https://www.rfc-editor.org/info/rfc8341>.

[RFC8407] Bierman, A., "Guidelines for Authors and Reviewers of Documents Containing YANG Data Models", BCP 216, RFC 8407, DOI 10.17487/RFC8407, October 2018, <https://www.rfc-editor.org/info/rfc8407>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

[RFC8525] Bierman, A., Bjorklund, M., Schoenwaelder, J., Watsen, K., and R. Wilton, "YANG Library", RFC 8525, DOI 10.17487/RFC8525, March 2019, <https://www.rfc-editor.org/info/rfc8525>.

[I-D.ietf-tcpm-rfc793bis]
Eddy, W. M., "Transmission Control Protocol (TCP) Specification", Work in Progress, Internet-Draft, draft-ietf-tcpm-rfc793bis-25, 7 September 2021, <https://www.ietf.org/archive/id/draft-ietf-tcpm-rfc793bis-25.txt>.

14.2. Informative References

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, DOI 10.17487/RFC2818, May 2000, <https://www.rfc-editor.org/info/rfc2818>.

[RFC3022] Srisuresh, P. and K. Egevang, "Traditional IP Network Address Translator (Traditional NAT)", RFC 3022, DOI 10.17487/RFC3022, January 2001, <https://www.rfc-editor.org/info/rfc3022>.

[RFC3444] Pras, A. and J. Schoenwaelder, "On the Difference between Information Models and Data Models", RFC 3444, DOI 10.17487/RFC3444, January 2003, <https://www.rfc-editor.org/info/rfc3444>.

[RFC3849] Huston, G., Lord, A., and P. Smith, "IPv6 Address Prefix Reserved for Documentation", RFC 3849, DOI 10.17487/RFC3849, July 2004, <https://www.rfc-editor.org/info/rfc3849>.

[RFC5737] Arkko, J., Cotton, M., and L. Vegoda, "IPv4 Address Blocks Reserved for Documentation", RFC 5737, DOI 10.17487/RFC5737, January 2010, <https://www.rfc-editor.org/info/rfc5737>.

[RFC8329] Lopez, D., Lopez, E., Dunbar, L., Strassner, J., and R. Kumar, "Framework for Interface to Network Security Functions", RFC 8329, DOI 10.17487/RFC8329, February 2018, <https://www.rfc-editor.org/info/rfc8329>.

[RFC8805] Kline, E., Duleba, K., Szamonek, Z., Moser, S., and W. Kumari, "A Format for Self-Published IP Geolocation Feeds", RFC 8805, DOI 10.17487/RFC8805, August 2020, <https://www.rfc-editor.org/info/rfc8805>.

[I-D.ietf-i2nsf-capability] Xia, L., Strassner, J., Basile, C., and D. R. Lopez, "Information Model of NSFs Capabilities", Work in Progress, Internet-Draft, draft-ietf-i2nsf-capability-05, 24 April 2019, <https://www.ietf.org/archive/id/draft-ietf-i2nsf-capability-05.txt>.

[IANA-ICMP-Parameters] Internet Assigned Numbers Authority (IANA), "Assigned Internet Protocol Numbers", February 2021, <https://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml>.

[IANA-ICMPv6-Parameters] Internet Assigned Numbers Authority (IANA), "Internet Control Message Procotol version 6 (ICMPv6) Parameters", February 2021, <https://www.iana.org/assignments/icmpv6-parameters/icmpv6-parameters.xhtml>.

[Encyclopedia-Britannica] Britannica, "Continent", September 2020, <https://www.britannica.com/science/continent>.

[YARA]
Alvarez, V., Bengen, H., Metz, J., Buehlmann, S., and W. Shields, "YARA", YARA Documents https://yara.readthedocs.io/en/v3.5.0/, August 2020.

[SURICATA] Julien, V. and ., "SURICATA", SURICATA Documents https://suricata-ids.org/docs/, August 2020.

[SNORT] Roesch, M., Green, C., and B. Caswell, "SNORT", SNORT Documents https://www.snort.org/#documents, August 2020.

[STIX] Jordan, B., Piazza, R., and T. Darley, "Structured Threat Information Expression (STIX)", STIX Version 2.1: Committee Specification 01 https://docs.oasis-open.org/cti/stix/v2.1/stix-v2.1.pdf, March 2020.

Appendix A. Changes from draft-ietf-i2nsf-consumer-facing-interface-dm-14

The following changes are made from draft-ietf-i2nsf-consumer-facing-interface-dm-14:

*This version has been updated following Tom Petch's comments.

Authors' Addresses

Jaehoon (Paul) Jeong (editor)
Department of Computer Science and Engineering
Sungkyunkwan University
2066 Seobu-Ro, Jangan-Gu
Suwon
Gyeonggi-Do
16419
Republic of Korea

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

Chaehong Chung
Department of Electronic, Electrical and Computer Engineering
Sungkyunkwan University
2066 Seobu-Ro, Jangan-Gu
Suwon
Gyeonggi-Do
16419
Republic of Korea

Phone: +82 31 299 4957
Email: darkhong@skku.edu

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

Rakesh Kumar
Juniper Networks
1133 Innovation Way
Sunnyvale, CA 94089
United States of America

Email: rkkumar@juniper.net

Susan Hares
Huawei
7453 Hickory Hill
Saline, MI 48176
United States of America

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