

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 4, 2017

S. Hares
Huawei
R. Moskowitz
HTT Consulting
L. Xia
Huawei
J. Kim
J. Jeong
Sungkyunkwan University
October 31, 2016

I2NSF Capability YANG Data Model
draft-hares-i2nsf-capability-data-model-00

Abstract

This document defines a YANG data model for capability that enables an I2NSF client to control various network security functions in network security devices via an I2NSF controller.

Status of This Memo

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

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

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

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

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

This Internet-Draft will expire on May 4, 2017.

Copyright Notice

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

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

Table of Contents

1.	Introduction	3
2.	Requirements Language	3
3.	Terminology	3
3.1.	Tree Diagrams	4
4.	High-Level YANG	4
4.1.	Capabilities per NSF	4
4.2.	Network Security Control	5
4.3.	Content Security Control	5
4.4.	Attack Mitigation Control	6
4.5.	IT Resources linked to Capabilities	8
4.6.	Actions	9
5.	YANG Modules	9
6.	IANA Considerations	26
7.	Security Considerations	26
8.	Acknowledgements	26
9.	References	27
9.1.	Normative References	27
9.2.	Informative References	27

1. Introduction

[i2nsf-problem-statement] proposes two different types of interfaces:

- o North-bound interface (NBI) provided by the network security functions (NSFs)
- o Interface between I2NSF user/client with network controller

This document provides a yang model that defines the capabilities for security devices that can be utilized by I2NSF NBI between the I2RS network controller and the NSF devices to express the NSF devices capabilities. It can also be used by the I2NSF user application (or I2NSF client) to network controller to provide a complete list of the I2NSF capabilities the Network controller can control. This document defines a YANG [RFC6020] data model based on the [i2nsf-cap-inf-im], and initial work done in [i2nsf-service-inf-dm]. Terms used in document are defined in [i2nsf-terminology]. [i2nsf-cap-inf-im] defines the following type of functionality in NSFs.

- o Network Security Control
- o Content Security Control
- o Attack Mitigation Control

This document contains high-level YANG for each type of control.

2. Requirements Language

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

3. Terminology

This document uses the terminology described in [[i2nsf-cap-inf-im](#)] [[i2rs-rib-data-model](#)] [[supa-policy-info-model](#)]. Especially, the following terms are from [[supa-policy-info-model](#)]:

- o Data Model: A data model is a representation of concepts of interest to an environment in a form that is dependent on data repository, data definition language, query language, implementation language, and protocol.
- o Information Model: An information model is a representation of concepts of interest to an environment in a form that is independent of data repository, data definition language, query

language, implementation language, and protocol.

[3.1.](#) Tree Diagrams

A simplified graphical representation of the data model is used in this document. The meaning of the symbols in these diagrams [[i2rs-rib-data-model](#)] is as follows:

- o Brackets "[" and "]" enclose list keys.
- o Abbreviations before data node names: "rw" means configuration (read-write) and "ro" state data (read-only).
- o Symbols after data node names: "?" means an optional node and "*" denotes a "list" and "leaf-list".
- o Parentheses enclose choice and case nodes, and case nodes are also marked with a colon (":").
- o Ellipsis ("...") stands for contents of subtrees that are not shown.

[4.](#) High-Level YANG

This section provides an overview of the high level YANG.

[4.1.](#) Capabilities per NSF

The high level YANG capabilities per NSF devices, controller, or

application is the following:

```
module : ietf-i2nsf-capability
  +--rw sec-ctl-capabilities
  +--rw nsf-capabilities
    +--rw nsf* [nsf-name]
      +--rw nsf-name string
      +--rw nsf-address inet:ipv4-address
      +--rw net-sec-control-capabilities
        | uses i2nsf-net-sec-control-caps
      +--rw con-sec-control-capabilities
        | uses i2nsf-con-sec-control-caps
      +--rw attack-mitigation-capabilities
        | uses i2nsf-attack-mitigation-control-caps
      +--rw it-resource
        | uses i2nsf-it-resources
```

Figure 1: High-Level YANG of I2NSF Capability Interface

Each of these section mirror sections in: [[i2nsf-cap-inf-im](#)]. The high level YANG for net-sec-control-capabilities, con-sec-control-capabilities, and attack-mitigation-capabilities. This draft is also utilizes the concepts originated in Basile, Lioy, Pitscheider, and Zhao[2015] concerning conflict resolution, use of external data, and IT-Resources. The authors are grateful to Cataldo for pointing out this excellent work.

[4.2.](#) Network Security Control

This section expands the

```
    +--rw net-sec-control-capabilities
      | uses i2nsf-net-sec-control-caps
```

Network Security Control

```
  +--rw i2nsf-net-sec-control-caps
    +--rw network-security-control
      +--rw nsc-support? boolean
      +--rw nsc-fcn* [nsc-fcn-name]
```

```
+--rw nsc-fcn-name  string  //std or vendor name
```

Figure 2: High-Level YANG of Network Security Control

4.3. Content Security Control

This section expands the

```
+--rw net-sec-control-capabilities
|  uses i2nsf-con-sec-control-caps
```

Content Security Control

```
+--rw i2nsf-con-sec-control-caps
+--rw content-security-control
+--rw antivirus
|  +--rw antivirus-support?  boolean
|  +--rw antivirus-fcn*  [antivirus-fcn-name]
|     +--rw antivirus-fcn-name  string  //std or vendor name
+--rw ips
|  +--rw ips-support?  boolean
|  +--rw ips-fcn*  [ips-fcn-name]
|     +--rw ips-fcn-name  string  //std or vendor name
+--rw ids
|  +--rw ids-support?  boolean
```

```
|  +--rw ids-fcn*  [ids-fcn-name]
|     +--rw ids-fcn-name  string  //std or vendor name
+--rw url-filter
|  +--rw url-filter-support?  boolean
|  +--rw url-filter-fcn*  [url-filter-fcn-name]
|     +--rw url-filter-fcn-name  string  //std or vendor name
+--rw data-filter
|  +--rw data-filter-support?  boolean
|  +--rw data-filter-fcn*  [data-filter-fcn-name]
|     +--rw data-filter-fcn-name  string  //std or vendor name
+--rw mail-filter
|  +--rw mail-filter-support?  boolean
|  +--rw mail-filter-fcn*  [mail-filter-fcn-name]
|     +--rw mail-filter-fcn-name  string  //std or vendor name
+--rw file-blocking
```

```

|   +---rw file-blocking-support?  boolean
|   +---rw file-blocking-fcn*   [file-blocking-fcn-name]
|     +---rw file-blocking-fcn-name  string //std or vendor name
+---rw file-isolate
|   +---rw file-isolate-support?  boolean
|   +---rw file-isolate-fcn*   [file-isolate-fcn-name]
|     +---rw file-isolate-fcn-name  string //std or vendor name
+---rw pkt-capture
|   +---rw pkt-capture-support?  boolean
|   +---rw pkt-capture-fcn*   [pkt-capture-fcn-name]
|     +---rw pkt-capture-fcn-name  string //std or vendor name
+---rw app-control
|   +---rw app-control-support?  boolean
|   +---rw app-control-fcn*   [app-control-fcn-name]
|     +---rw app-control-fcn-name  string //std or vendor name
+---rw voip-volte
    +---rw voip-volte-support?  boolean
    +---rw voip-volte-fcn*   [voip-volte-fcn-name]
        +---rw voip-volte-fcn-name  string //std or vendor name

```

Figure 3: High-Level YANG of Content Security Control

4.4. Attack Mitigation Control

This high level YANG below expands the following section of the top-level model:

```

+---rw attack-mitigation-control-capabilities
|   uses i2nsf-attack-mitigation-control-caps

```

Attack Mitigation Control

```

+---rw i2nsf-attack-mitigation-control-caps

```

```

+---rw attack-mitigation-control
  +---rw (attack-mitigation-control-type)?
    +---: (ddos-attack)
      |   +---rw (ddos-attack-type)?
      |     +---: (network-layer-ddos-attack)
      |       |   +---rw network-layer-ddos-attack-types
      |       |     +---rw syn-flood-attack
      |       |       |   +---rw syn-flood-attack-support?  boolean

```

```

| | | +--rw syn-flood-fcn* [syn-flood-fcn-name]
| | |   +--rw syn-flood-fcn-name string
+--rw udp-flood-attack
| +--rw udp-flood-attack-support? boolean
| +--rw udp-flood-fcn* [udp-flood-fcn-name]
|   +--rw udp-flood-fcn-name string
+--rw icmp-flood-attack
| +--rw icmp-flood-attack-support? boolean
| +--rw icmp-flood-fcn* [icmp-flood-fcn-name]
|   +--rw icmp-flood-fcn-name string
+--rw ip-fragment-flood-attack
| +--rw ip-fragment-flood-attack-support? boolean
| +--rw ip-frag-flood-fcn* [ip-frag-flood-fcn-name]
|   +--rw ip-frag-flood-fcn-name string
+--rw ipv6-related-attack
| +--rw ipv6-related-attack-support? boolean
| +--rw ipv6-related-fcn* [ipv6-related-fcn-name]
|   +--rw ipv6-related-fcn-name string
+--: (app-layer-ddos-attack)
  +--rw app-layer-ddos-attack-types
    +--rw http-flood-attack
    | +--rw http-flood-attack-support? boolean
    | +--rw http-flood-fcn* [http-flood-fcn-name]
    |   +--rw http-flood-fcn-name string
    +--rw https-flood-attack
    | +--rw https-flood-attack-support? boolean
    | +--rw https-flood-fcn* [https-flood-fcn-name]
    |   +--rw https-flood-fcn-name string
    +--rw dns-flood-attack
    | +--rw dns-flood-attack-support? boolean
    | +--rw dns-flood-fcn* [dns-flood-fcn-name]
    |   +--rw dns-flood-fcn-name string
    +--rw dns-amp-flood-attack
    | +--rw dns-amp-flood-attack-support? boolean
    | +--rw dns-amp-flood-fcn* [dns-amp-flood-fcn-name]
    |   +--rw dns-amp-flood-fcn-name string
    +--rw ssl-ddos-attack
    | +--rw ssl-ddos-attack-support? boolean
    | +--rw ssl-ddos-fcn* [ssl-ddos-fcn-name]
    |   +--rw ssl-ddos-fcn-name string

```

+--: (single-packet-attack)


```

+--rw (single-packet-attack-type)?
+--: (scan-and-sniff-attack)
|   +--rw ip-sweep-attack
|   |   +--rw ip-sweep-attack-support?  boolean
|   |   +--rw ip-sweep-fcn*  [ip-sweep-fcn-name]
|   |   +--rw ip-sweep-fcn-name  string
|   +--rw port-scanning-attack
|   |   +--rw port-scanning-attack-support?  boolean
|   |   +--rw port-scanning-fcn*  [port-scanning-fcn-name]
|   |   +--rw port-scanning-fcn-name  string
+--: (malformed-packet-attack)
|   +--rw ping-of-death-attack
|   |   +--rw ping-of-death-attack-support?  boolean
|   |   +--rw ping-of-death-fcn*  [ping-of-death-fcn-name]
|   |   +--rw ping-of-death-fcn-name  string
|   +--rw teardrop-attack
|   |   +--rw teardrop-attack-support?  boolean
|   |   +--rw tear-drop-fcn*  [tear-drop-fcn-name]
|   |   +--rw tear-drop-fcn-name  string
+--: (special-packet-attack)
+--rw oversized-icmp-attack
|   +--rw oversized-icmp-attack-support?  boolean
|   +--rw oversized-icmp-fcn*  [oversized-icmp-fcn-name]
|   +--rw oversized-icmp-fcn-name  string
+--rw tracert-attack
+--rw tracert-attack-support?  boolean
+--rw tracert-fcn*  [tracert-fcn-name]
+--rw tracert-fcn-name  string

```

Figure 4: High-Level YANG of Attack Mitigation Control

[4.5.](#) IT Resources linked to Capabilities

This section provides a link between capabilities and IT resources. This section has a list of IT resources by name. Additional input is needed.

```
+--rw it-resource
| uses i2nsf-it-resources
```

It Resource

```
+--rw i2nsf-it-resources
  +--rw it-resources* [it-resource-id]
    +--rw it-resource-id uint64
    +--rw it-resource-name string
```

Figure 5: High-Level YANG of IT Resources

4.6. Actions

Notifications indicate when rules are added or deleted. These notifications will be defined later.

5. YANG Modules

This section introduces a YANG module for the information model of I2NSF capability interface, as defined in the [[i2nsf-cap-inf-im](#)].

```
<CODE BEGINS> file "ietf-i2nsf-capability@2016-10-31.yang"
```

```
module ietf-i2nsf-capability {
  namespace
    "urn:ietf:params:xml:ns:yang:ietf-i2nsf-capability";
  prefix
    i2nsf-capability;

  import ietf-inet-types{
    prefix inet;
  }

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

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

    WG Chair: Adrian Farrel
    <mailto:Adrain@olddog.co.uk>
```

Internet-Draft

I2NSF Capability YANG Data Model

October 2016

<mailto:Linda.dunbar@huawei.com>

Editor: Susan Hares
<mailto:shares@ndzh.com>

Editor: Jinyong Tim Kim
<mailto:wlsdyd0930@nate.com>

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

description

"This module describes a capability model
for I2NSF devices.";

revision "2016-10-31"{

description "Third revision";

reference

["draft-xia-i2nsf-capability-interface-im-06
draft-hares-i2nsf-capability-yang-01"](#);

}

container sec-ctl-capabilities {

description

"sec-ctl-capabilities";

}

grouping i2nsf-net-sec-control-caps {

description

"i2nsf-net-sec-control-caps";

container network-security-control {

description

"i2nsf-net-sec-control-caps";

leaf nsc-support {

type boolean;

mandatory true;

description

```
        "nsc-support";
    }
    list nsc-fcn {
        key "nsc-fcn-name";
        description
            "nsc-fcn";
        leaf nsc-fcn-name {
            type string;
            mandatory true;
        }
    }
}
```

```
        description
            "nsc-fcn-name";
    }
}
}
}

grouping i2nsf-con-sec-control-caps {
    description
        "i2nsf-con-sec-control-caps";

    container content-security-control {
        description
            "content-security-control";

        container antivirus {
            description
                "antivirus";

            leaf antivirus-support {
                type boolean;
                mandatory true;
                description
                    "antivirus-support";
            }
            list antivirus-fcn-name {
                key "antivirus-fcn-name";
                description
                    "antivirus-fcn-name";

                leaf antivirus-fcn-name {
                    type string;
                }
            }
        }
    }
}
```

```
        mandatory true;
        description
            "antivirus-fcn-name";
    }
}
}
```

```
container ips {
    description
        "ips";

    leaf ips-support {
        type boolean;
        mandatory true;
        description
            "ips-support";
    }
}
```

```
    }
    list ips-fcn {
        key "ips-fcn-name";
        description
            "ips-fcn";

        leaf ips-fcn-name {
            type string;
            mandatory true;
            description
                "ips-fcn-name";
        }
    }
}
}
```

```
container ids {
    description
        "ids";

    leaf ids-support {
        type boolean;
        mandatory true;
        description
            "ids-support";
    }
}
```

```

list ids-fcn {
  key "ids-fcn-name";
  description
    "ids-fcn";

  leaf ids-fcn-name {
    type string;
    mandatory true;
    description
      "ids-fcn-name";
  }
}

container url-filter {
  description
    "url-filter";

  leaf url-filter-support {
    type boolean;
    mandatory true;
    description
      "url-filter-support";
  }
}

```

```

}
list url-filter-fcn {
  key "url-filter-fcn-name";
  description
    "url-filter-fcn";

  leaf url-filter-fcn-name {
    type string;
    mandatory true;
    description
      "url-filter-fcn-name";
  }
}

container data-filter {
  description
    "data-filter";
}

```

```

leaf data-filter-support {
  type boolean;
  mandatory true;
  description
    "data-filter-support";
}
list data-filter-fcn {
  key "data-filter-fcn-name";
  description
    "data-filter-fcn";

  leaf data-filter-fcn-name {
    type string;
    mandatory true;
    description
      "data-filter-fcn-name";
  }
}
}
}

```

```

container mail-filter {
  description
    "mail-filter";

  leaf mail-filter-support {
    type boolean;
    mandatory true;
    description
      "mail-filter-support";
  }
}

```

```

}
list mail-filter-fcn {
  key "mail-filter-fcn-name";
  description
    "mail-filter-fcn";

  leaf mail-filter-fcn-name {
    type string;
    mandatory true;
    description
      "mail-filter-fcn-name";
  }
}

```

```

    }
  }
}

container file-blocking {
  description
    "file-blocking";

  leaf file-blocking-support {
    type boolean;
    mandatory true;
    description
      "file-blocking-support";
  }
  list file-blocking-fcn {
    key "file-blocking-fcn-name";
    description
      "file-blocking-fcn";

    leaf file-blocking-fcn-name {
      type string;
      mandatory true;
      description
        "file-blocking-fcn-name";
    }
  }
}

```

```

container file-isolate {
  description
    "file-isolate";

  leaf file-isolate-support {
    type boolean;
    mandatory true;
    description
      "file-isolate-support";
  }
}

```

```

}
list file-isolate-fcn {
  key "file-isolate-fcn-name";
  description

```



```

        "file-isolate-fcn";

    leaf file-isolate-fcn-name {
        type string;
        mandatory true;
        description
            "file-isolate-fcn-name";
    }
}

container pkt-capture {
    description
        "pkt-capture";

    leaf pkt-capture-support {
        type boolean;
        mandatory true;
        description
            "pkt-capture-support";
    }
    list pkt-capture-fcn {
        key "pkt-capture-fcn-name";
        description
            "pkt-capture-fcn";

        leaf pkt-capture-fcn-name {
            type string;
            mandatory true;
            description
                "pkt-capture-fcn-name";
        }
    }
}

container app-control {
    description
        "app-control";

    leaf app-control-support {
        type boolean;
        mandatory true;
        description
            "app-control-support";
    }
}

```

```
    }
    list app-control-fcn {
      key "app-control-fcn-name";
      description
        "app-control-fcn";

      leaf app-control-fcn-name {
        type string;
        mandatory true;
        description
          "app-control-fcn-name";
      }
    }
  }
}

container voip-volte {
  description
    "voip-volte";

  leaf voip-volte-support {
    type boolean;
    mandatory true;
    description
      "voip-volte-support";
  }
  list voip-volte-fcn {
    key "voip-volte-fcn-name";
    description
      "voip-volte-fcn";

    leaf voip-volte-fcn-name {
      type string;
      mandatory true;
      description
        "voip-volte-fcn-name";
    }
  }
}

}

grouping i2nsf-attack-mitigation-control-caps {
  description
    "i2nsf-attack-mitigation-control-caps";

  container attack-mitigation-control {
    description
```

"attack-mitigation-control";

```
choice attack-mitigation-control-type {
  description
    "attack-mitigation-control-type";
  case ddos-attack {
    description
      "ddos-attack";
    choice ddos-attack-type {
      description
        "ddos-attack-type";
      case network-layer-ddos-attack {
        description
          "network-layer-ddos-attack";
        container network-layer-ddos-attack-types {
          description
            "network-layer-ddos-attack-type";
          container syn-flood-attack {
            description
              "syn-flood-attack";
            leaf syn-flood-attack-support {
              type boolean;
              mandatory true;
              description
                "syn-flood-attack-support";
            }
          }
          list syn-flood-fcn {
            key "syn-flood-fcn-name";
            description
              "syn-flood-fcn";
            leaf syn-flood-fcn-name {
              type string;
              mandatory true;
              description
                "syn-flood-fcn-name";
            }
          }
        }
      }
    }
  container udp-flood-attack {
    description
      "udp-flood-attack";
    leaf udp-flood-attack-support {
```

```
    type boolean;
    mandatory true;
    description
      "udp-flood-attack-support";
  }
  list udp-flood-fcn {
    key "udp-flood-fcn-name";
    description
```

```
    "udp-flood-fcn";
    leaf udp-flood-fcn-name {
      type string;
      mandatory true;
      description
        "udp-flood-fcn-name";
    }
  }
}
container icmp-flood-attack {
  description
    "icmp-flood-attack";
  leaf icmp-flood-attack-support {
    type boolean;
    mandatory true;
    description
      "icmp-flood-attack-support";
  }
  list icmp-flood-fcn {
    key "icmp-flood-fcn-name";
    description
      "icmp-flood-fcn";
    leaf icmp-flood-fcn-name {
      type string;
      mandatory true;
      description
        "icmp-flood-fcn-name";
    }
  }
}
container ip-fragment-flood-attack {
  description
    "ip-fragment-flood-attack";
```

```

leaf ip-fragment-flood-attack-support {
  type boolean;
  mandatory true;
  description
    "ip-fragment-flood-attack-support";
}
list frag-flood-fcn {
  key "ip-frag-flood-fcn-name";
  description
    "frag-flood-fcn";
  leaf ip-frag-flood-fcn-name {
    type string;
    mandatory true;
    description
      "ip-frag-flood-fcn-name";
  }
}

```

```

    }
  }
}
container ipv6-related-attack {
  description
    "ipv6-related-attack";
  leaf ipv6-related-attack-support {
    type boolean;
    mandatory true;
    description
      "ipv6-related-attack-support";
  }
  list ipv6-related-fcn {
    key "ipv6-related-fcn-name";
    description
      "ipv6-related-fcn";
    leaf ipv6-related-fcn-name {
      type string;
      mandatory true;
      description
        "ipv6-related-fcn-name";
    }
  }
}
}
}
}

```

```

case app-layer-ddos-attack {
  description
    "app-layer-ddos-attack";
  container app-layer-ddos-attack-types {
    description
      "app-layer-ddos-attack-types";
    container http-flood-attack {
      description
        "http-flood-attack";
      leaf http-flood-attack-support {
        type boolean;
        mandatory true;
        description
          "http-flood-attack-support";
      }
    }
    list http-flood-fcn {
      key "http-flood-fcn-name";
      description
        "http-flood-fcn";
      leaf http-flood-fcn-name {
        type string;
        mandatory true;
      }
    }
  }
}

```

```

      description
        "http-flood-fcn-name";
    }
  }
}
container https-flood-attack {
  description
    "https-flood-attack";
  leaf https-flood-attack-support {
    type boolean;
    mandatory true;
    description
      "https-flood-attack-support";
  }
  list https-flood-fcn {
    key "https-flood-fcn-name";
    description
      "https-flood-fcn";
    leaf https-flood-fcn-name {

```

```

        type string;
        mandatory true;
        description
            "https-flood-fcn-name";
    }
}
}
container dns-flood-attack {
    description
        "dns-flood-attack";
    leaf dns-flood-attack-support {
        type boolean;
        mandatory true;
        description
            "dns-flood-attack-support";
    }
    list dns-flood-fcn {
        key "dns-flood-fcn-name";
        description
            "dns-flood-fcn";
        leaf dns-flood-fcn-name {
            type string;
            mandatory true;
            description
                "dns-flood-fcn-name";
        }
    }
}
}
container dns-amp-flood-attack {

```

```

    description
        "dns-amp-flood-attack";
    leaf dns-flood-attack-support {
        type boolean;
        mandatory true;
        description
            "dns-flood-attack-support";
    }
}
list dns-amp-flood-fcn {
    key "dns-amp-flood-fcn-name";
    description
        "dns-amp-flood-fcn";

```

```

        leaf dns-amp-flood-fcn-name {
            type string;
            mandatory true;
            description
                "dns-amp-flood-fcn-name";
        }
    }
}
container ssl-ddos-attack {
    description
        "ssl-ddos-attack";
    leaf ssl-ddos-attack-support {
        type boolean;
        mandatory true;
        description
            "ssl-ddos-attack-support";
    }
    list ssl-ddos-fcn {
        key "ssl-ddos-fcn-name";
        description
            "ssl-ddos-fcn";
        leaf ssl-ddos-fcn-name {
            type string;
            mandatory true;
            description
                "ssl-ddos-fcn-name";
        }
    }
}
}
}
}
}
}
}
}
}

case single-packet-attack {
    description

```

```

        "single-packet-attack";
    choice single-packet-attack-type {
        description
            "single-packet-attack-type";
        case scan-and-sniff-attack {

```



```

description
  "scan-and-sniff-attack";
container ip-sweep-attack {
  description
    "ip-sweep-attack";
  leaf ip-sweep-attack-suppor {
    type boolean;
    mandatory true;
    description
      "ip-sweep-attack-suppor";
  }
  list ip-sweep-fcn {
    key "ip-sweep-fcn-name";
    description
      "ip-sweep-fcn";
    leaf ip-sweep-fcn-name {
      type string;
      mandatory true;
      description
        "ip-sweep-fcn-name";
    }
  }
}
container port-scanning-attack {
  description
    "port-scanning-attack";
  leaf port-scanning-attack-support {
    type boolean;
    mandatory true;
    description
      "port-scanning-attack-support";
  }
  list port-scanning-fcn {
    key "port-scanning-fcn-name";
    description
      "port-scanning-fcn";
    leaf port-scanning-fcn-name {
      type string;
      mandatory true;
      description
        "port-scanning-fcn-name";
    }
  }
}

```

```

    }
  }
  case malformed-packet-attack {
    description
      "malformed-packet-attack";
    container ping-of-death-attack {
      description
        "ping-of-death-attack";
      leaf ping-of-death-attack-support {
        type boolean;
        mandatory true;
        description
          "ping-of-death-attack-support";
      }
      list ping-of-death-fcn {
        key "ping-of-death-fcn-name";
        description
          "ping-of-death-fcn";
        leaf ping-of-death-fcn-name {
          type string;
          mandatory true;
          description
            "ping-of-death-fcn-name";
        }
      }
    }
  }
  container teardrop-attack {
    description
      "teardrop-attack";
    leaf teardrop-attack-support {
      type boolean;
      mandatory true;
      description
        "teardrop-attack-support";
    }
    list tear-drop-fcn {
      key "tear-drop-fcn-name";
      description
        "tear-drop-fcn";
      leaf tear-drop-fcn-name {
        type string;
        mandatory true;
        description
          "tear-drop-fcn-name";
      }
    }
  }
}

```

```
case special-packet-attack {
  description
    "special-packet-attack";
  container oversized-icmp-attack {
    description
      "oversized-icmp-attack";
    leaf oversized-icmp-attack-support {
      type boolean;
      mandatory true;
      description
        "oversized-icmp-attack-support";
    }
    list oversized-icmp-fcn {
      key "oversized-icmp-fcn-name";
      description
        "oversized-icmp-fcn";
      leaf oversized-icmp-fcn-name {
        type string;
        mandatory true;
        description
          "oversized-icmp-fcn-name";
      }
    }
  }
}
container tracert-attack {
  description
    "tracert-attack";
  leaf tracert-attack-support {
    type boolean;
    mandatory true;
    description
      "tracert-attack-support";
  }
  list tracert-fcn {
    key "tracert-fcn-name";
    description
      "tracert-fcn";
    leaf tracert-fcn-name {
      type string;
      mandatory true;
      description
        "tracert-fcn-name";
    }
  }
}
```

```
    }
  }
}
}
```

```
    }
  }
}

grouping i2nsf-it-resources {
  description
    "i2nsf-it-resource";
  list it-resources {
    key "it-resource-id";
    description
      "it-resource";
    leaf it-resource-id {
      type uint64;
      mandatory true;
      description
        "it-resource-id";
    }
    leaf it-resource-name {
      type string;
      mandatory true;
      description
        "it-resource-name";
    }
  }
}

container nsf-capabilities {
  description
    "nsf-capabilities";

  list nsf {
    key "nsf-name";
    description
      "nsf";
    leaf nsf-name {
```

```
        type string;
        mandatory true;
        description
            "nsf-name";
    }
    leaf nsf-address {
        type inet:ipv4-address;
        mandatory true;
        description
            "nsf-address";
    }
    container net-sec-control-capabilities {
        uses i2nsf-net-sec-control-caps;
    }
}
```

```
        description
            "net-sec-control-capabilities";
    }
    container con-sec-control-capabilities {
        uses i2nsf-con-sec-control-caps;
        description
            "con-sec-control-capabilities";
    }
    container attack-mitigation-capabilities {
        uses i2nsf-attack-mitigation-control-caps;
        description
            "attack-mitigation-capabilities";
    }
    container it-resource {
        uses i2nsf-it-resources;
        description
            "it-resource";
    }
}
}
```

<CODE ENDS>

Figure 6: Data Model of I2NSF Capability Interface

[6.](#) IANA Considerations

No IANA considerations exist for this document at this time. URL will be added.

[7.](#) Security Considerations

This document introduces no additional security threats and SHOULD follow the security requirements as stated in [[i2nsf-framework](#)].

[8.](#) Acknowledgements

This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (No.R-20160222-002755, Cloud based Security Intelligence Technology Development for the Customized Security Service Provisioning).

This document has greatly benefited from inputs by Daeyoung Hyun, Hyounghick Kim, Jung-Soo Park, Tae-Jin Ahn, and Se-Hui Lee.

Hares, et al.

Expires May 4, 2017

[Page 26]

Internet-Draft

I2NSF Capability YANG Data Model

October 2016

[9.](#) References

[9.1.](#) Normative References

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

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

[9.2.](#) Informative References

[i2nsf-cap-inf-im] Xia, L., Strassner, J., Li, K., Zhang, D., Lopez, E., BOUTHORS, N., and L. Fang, "Information Model of Interface to Network Security Functions Capability Interface", [draft-xia-i2nsf-capability-interface-im-06](#) (work in progress), June 2016.

- [i2nsf-problem-statement] Hares, S., Dunbar, L., Lopez, D., Zarny, M., and C. Jacquenet, "I2NSF Problem Statement and Use cases", [draft-ietf-i2nsf-problem-and-use-cases-02](#) (work in progress), October 2016.
- [i2nsf-service-inf-dm] Xia, L., Strassner, J., and D. Bogdanovic, "Data Model of Interface to Network Security Functions Service Interface", [draft-xia-i2nsf-service-interface-dm-00](#) (work in progress), February 2015.
- [i2nsf-terminology] Hares, S., Strassner, J., Lopez, D., Xia, L., and H. Birkholz, "Interface to Network Security Functions (I2NSF) Terminology", [draft-ietf-i2nsf-terminology-02](#) (work in progress), October 2016.
- [i2rs-rib-data-model] Wang, L., Ananthakrishnan, H., Chen, M., Dass, A., Kini, S., and N. Bahadur, "A YANG Data Model for Routing Information Base (RIB)", [draft-ietf-i2rs-rib-data-model-06](#) (work in progress), July 2016.
- [supa-policy-info-model] Strassner, J., halpern, j., and S. van der

- Meer, "Generic Policy Information Model for Simplified Use of Policy Abstractions (SUPA)", [draft-ietf-supa-generic-policy-info-model-01](#) (work in progress), July 2016.
- [i2nsf-framework] Lopez, D., Lopez, E., Dunbar, L., Strassner, J., and R. Kumar, "Framework for Interface to Network Security Functions", [draft-ietf-i2nsf-framework-04](#) (work in progress), October 2016.

Susan Hares
Huawei
7453 Hickory Hill
Saline, MI 48176
USA

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

Robert Moskowitz
HTT Consulting
Oak Park, MI
USA

Phone: +1-248-968-9809
EMail: rgm@htt-consult.com

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

Phone:
EMail: Frank.xialiang@huawei.com

Jinyong Tim Kim
Department of Computer Engineering
Sungkyunkwan University
2066 Seobu-Ro, Jangan-Gu
Suwon, Gyeonggi-Do 16419
Republic of Korea

Phone: +82 10 8273 0930
EMail: wlsdyd0930@nate.com

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

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