Abstract

This document defines a YANG data model for the capabilities of various Network Security Functions (NSFs) in the Interface to Network Security Functions (I2NSF) framework to centrally manage the capabilities of the various NSFs.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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

This Internet-Draft will expire on January 26, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents.
1. Introduction

As the industry becomes more sophisticated and network devices (e.g., Internet of Things, Self-driving vehicles, and VoIP/VoLTE smartphones), service providers have a lot of problems described in [RFC8192]. To resolve these problems, [draft-ietf-i2nsf-capability] specifies the information model of the capabilities of Network Security Functions (NSFs).
This document provides a YANG data model [RFC6020][RFC7950] that defines the capabilities of NSFs to centrally manage the capabilities of those security devices. The security devices can register their own capabilities into a Network Operator Management (Mgmt) System (i.e., Security Controller) with this YANG data model through the registration interface [RFC8329]. With the capabilities of those security devices maintained centrally, those security devices can be easily managed [RFC8329]. This YANG data model is based on the information model for I2NSF NSF capabilities [draft-ietf-i2nsf-capability].

This YANG data model uses an "Event-Condition-Action" (ECA) policy model that is used as the basis for the design of I2NSF Policy as described in [RFC8329] and [draft-ietf-i2nsf-capability]. The "ietf-i2nsf-capability" YANG module defined in this document provides the following features:

- Definition for general capabilities of network security functions.
- Definition for event capabilities of generic network security functions.
- Definition for condition capabilities of generic network security functions.
- Definition for condition capabilities of advanced network security functions.
- Definition for action capabilities of generic network security functions.
- Definition for resolution strategy capabilities of generic network security functions.
- Definition for default action capabilities of generic network security functions.

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][RFC8174].

3. Terminology

This document uses the terminology described in [draft-ietf-i2nsf-terminology][draft-ietf-i2nsf-capability][RFC8431][draft-ietf-supa-generic-policy-info-model]. Especially,
the following terms are from
[draft-ietf-supa-generic-policy-info-model]:

- **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.

- **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 is referred from [RFC8340].

### 4. Overview

This section provides an overview of how the YANG data model can be used in the I2NSF framework described in [RFC8329]. Figure 1 shows the capabilities of NSFs in I2NSF Framework. As shown in this figure, an NSF Developer’s Mgmt System can register NSFs and the capabilities that the network security device can support. To register NSFs in this way, the Developer’s Mgmt System utilizes this standardized capabilities YANG data model through its registration interface. With the capabilities of those network security devices maintained centrally, those security devices can be easily managed, which can resolve many of the problems described in [RFC8192]. The use cases are described below.

Note that the NSF-Facing Interface is used to configure the security policy rules of the generic network security functions [draft-ietf-i2nsf-nsf-facing-interface-dm], and the NSF Monitoring Interface is used to configure the security policy rules of advanced network security functions [draft-dong-i2nsf-asf-config], respectively, according to the capabilities of NSFs registered with the I2NSF Framework.
If a network manager wants to apply security policy rules to block malicious users, it is a tremendous burden to apply all of the needed rules to NSFs one-by-one. This problem can be resolved by managing the capabilities of NSFs. If network manager wants to block malicious users with IPv6, the network manager sends the security policy rules to block the users to the Network Operator Mgmt System using I2NSF user (i.e., a web browser or a software). When the Network Operator Mgmt System receives the security policy rules, it automatically sends that security policy rules to appropriate NSFs (i.e., NSF-m in Developer’s Mgmt System A and NSF-1 in Developer’s Mgmt System B) which can support the capabilities (i.e., IPv6). Therefore, an I2NSF User need not consider NSFs where to which NSFs the rules apply.

If NSFs encounter the malicious packets, it is a tremendous burden for the network manager to apply the rule to block the malicious packets to NSFs one-by-one. This problem can be resolved by
managing the capabilities of NSFs. If NSFs encounter the suspicious IPv4 packets, they can ask the Network Operator Mgmt System for information about the suspicious IPv4 packets in order to alter specific rules and/or configurations. When the Network Operator Mgmt System receives information, it inspects the information about the suspicious IPv4 packets. If the suspicious packets are determined to be malicious packets, the Network Operator Mgmt System creates and sends the security policy rules blocking malicious packets to appropriate NSFs (i.e., NSF-1 in Developer’s Mgmt System A and NSF-1 and NSF-n in Developer’s Mgmt System B) which can support the capabilities (i.e., IPv4). Therefore, the new security policy rules blocking malicious packets can be applied to appropriate NSFs without humans intervention.

5. YANG Tree Diagram

This section shows an YANG tree diagram of capabilities for network security functions, as defined in the [draft-ietf-i2nsf-capability].

5.1. Network Security Function (NSF) Capabilities

This section shows YANG tree diagram for NSF capabilities.
This YANG tree diagram shows NSF capabilities.

The model includes NSF capabilities. The NSF capabilities include time capabilities, event capabilities, condition capabilities, action capabilities, resolution strategy capabilities, and default action capabilities.

Time capabilities are used to specify the capabilities to specify when to execute the I2NSF policy rule. The time capabilities are defined in terms of absolute time and periodic time. The absolute time means the exact time to start or end. The periodic time means repeated time like day, week, or month.

Event capabilities are used to specify capabilities how to trigger the evaluation of the condition clause of the I2NSF Policy Rule. The
defined event capabilities are defined as system event and system alarm. The event capability can be extended according to specific vendor condition features. The event capability is described in detail in [draft-ietf-i2nsf-capability].

Condition capabilities are used to specify capabilities of a set of attributes, features, and/or values that are to be compared with a set of known attributes, features, and/or values in order to determine whether or not the set of actions in that (imperative) I2NSF policy rule can be executed. The condition capabilities are classified in terms of generic network security functions and advanced network security functions. The condition capabilities of generic network security functions are defined as IPv4 capability, IPv6 capability, TCP capability, UDP capability, and ICMP capability. The condition capabilities of advanced network security functions are defined as anti-virus capability, anti-ddos capability, IPS capability, HTTP capability, and VoIP/VoLTE capability. The condition capability can be extended according to specific vendor condition features. The condition capability is described in detail in [draft-ietf-i2nsf-capability].

Action capabilities are used to specify capabilities of how to control and monitor aspects of flow-based NSFs when the event and condition clauses are satisfied. The action capabilities are defined as ingress-action capability, egress-action capability, and log-action capability. The action capability can be extended according to specific vendor action features. The action capability is described in detail in [draft-ietf-i2nsf-capability].

Resolution strategy capabilities are used to specify capabilities of how to resolve conflicts that occur between the actions of the same or different policy rules that are matched and contained in this particular NSF. The resolution strategy capabilities are defined as First Matching Rule (FMR), Last Matching Rule (LMR), Prioritized Matching Rule (PMR), Prioritized Matching Rule with Errors (PMRE), and Prioritized Matching Rule with No Errors (PMRN). The resolution strategy capabilities can be extended according to specific vendor action features. The resolution strategy capability is described in detail in [draft-ietf-i2nsf-capability].

Default action capabilities are used to specify capabilities of how to execute I2NSF policy rules when no rule matches a packet. The default action capabilities are defined as pass, drop, reject, alert, and mirror. The default action capability can be extended according to specific vendor action features. The default action capability is described in detail in [draft-ietf-i2nsf-capability].
IPsec method capabilities are used to specify capabilities of how to support an Internet Key Exchange (IKE) for the security communication. The default action capabilities are defined as IKE and IKE-less. The default action capability can be extended according to specific vendor action features. The default action capability is described in detail in [draft-ietf-i2nsf-sdn-ipsec-flow-protection].

6. YANG Data Modules

6.1. I2NSF Capability YANG Data Module

This section introduces a YANG data module for network security functions capabilities, as defined in the [draft-ietf-i2nsf-capability].

<CODE BEGINS> file "ietf-i2nsf-capability@2019-07-24.yang"

module ietf-i2nsf-capability {
  yang-version 1.1;
  namespace
  prefix nsfcap;

  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: Linda Dunbar
    <mailto:ldunbar@futurewei.com>

    WG Chair: Yoav Nir
    <mailto:ynir.ietf@gmail.com>

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

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

    Editor: Jinyong Tim Kim
    <mailto:timkim@skku.edu>";

</CODE ENDS>
description
"This module describes a capability model for I2NSF devices.

Copyright (c) 2018 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 (http://trustee.ietf.org/license-info).

This version of this YANG module is part of RFC 8341; see the RFC itself for full legal notices.";

revision "2019-07-24"{
  description "Initial revision.";
  reference
    "RFC XXXX: I2NSF Capability YANG Data Model";
}

/*
 * Identities
 */

identity event {
  description
    "Base identity for I2NSF policy events.";
  reference
    "draft-ietf-i2nsf-nsf-monitoring-data-model-01
     - Event";
}

identity system-event-capability {
  base event;
  description
    "Identity for system events";
  reference
    "draft-ietf-i2nsf-nsf-monitoring-data-model-01
     - System alarm";
}

identity system-alarm-capability {
  base event;
  description
    "Identity for system alarms";
reference

"draft-ietf-i2nsf-nsf-monitoring-data-model-01
 - System alarm";
}

identity access-violation {
  base system-event-capability;
  description
    "Identity for access violation events";
  reference
    "draft-ietf-i2nsf-nsf-monitoring-data-model-01
     - System event";
}

identity configuration-change {
  base system-event-capability;
  description
    "Identity for configuration change events";
  reference
    "draft-ietf-i2nsf-nsf-monitoring-data-model-01
     - System event";
}

identity memory-alarm {
  base system-alarm-capability;
  description
    "Identity for memory alarm events";
  reference
    "draft-ietf-i2nsf-nsf-monitoring-data-model-01
     - System alarm";
}

identity cpu-alarm {
  base system-alarm-capability;
  description
    "Identity for CPU alarm events";
  reference
    "draft-ietf-i2nsf-nsf-monitoring-data-model-01
     - System alarm";
}

identity disk-alarm {
  base system-alarm-capability;
  description
    "Identity for disk alarm events";
  reference
    "draft-ietf-i2nsf-nsf-monitoring-data-model-01
     - System alarm";
identity hardware-alarm {
    base system-alarm-capability;
    description "Identity for hardware alarm events";
    reference "draft-ietf-i2nsf-nsf-monitoring-data-model-01 - System alarm";
}

identity interface-alarm {
    base system-alarm-capability;
    description "Identity for interface alarm events";
    reference "draft-ietf-i2nsf-nsf-monitoring-data-model-01 - System alarm";
}

identity condition {
    description "Base identity for policy conditions";
}

identity context-capability {
    base condition;
    description "Identity for context condition capabilities";
}

identity acl-number {
    base context-capability;
    description "Identity for ACL number condition capability";
}

identity application {
    base context-capability;
    description "Identity for application condition capability";
}

identity target {
    base context-capability;
    description "Identity for target condition capability";
}
identity user {
  base context-capability;
  description
      "Identity for user condition capability";
}

identity group {
  base context-capability;
  description
      "Identity for group condition capability";
}

identity geography {
  base context-capability;
  description
      "Identity for geography condition capability";
}

identity ipv4-capability {
  base condition;
  description
      "Identity for IPv4 condition capabilities";
  reference
      "RFC 791: Internet Protocol";
}

identity exact-ipv4-header-length {
  base ipv4-capability;
  description
      "Identity for exact-match IPv4 header-length condition capability";
  reference
      "RFC 791: Internet Protocol - Header Length";
}

identity range-ipv4-header-length {
  base ipv4-capability;
  description
      "Identity for range-match IPv4 header-length condition capability";
  reference
      "RFC 791: Internet Protocol - Header Length";
}

identity ipv4-tos {
  base ipv4-capability;
  description
      "Identity for IPv4 Type-Of-Service (TOS)
condition capability";
reference
 "RFC 791: Internet Protocol - Type of Service";
}

identity exact-ipv4-total-length {
  base ipv4-capability;
  description
    "Identity for exact-match IPv4 total length condition capability";
  reference
    "RFC 791: Internet Protocol - Total Length";
}

identity range-ipv4-total-length {
  base ipv4-capability;
  description
    "Identity for range-match IPv4 total length condition capability";
  reference
    "RFC 791: Internet Protocol - Total Length";
}

identity ipv4-id {
  base ipv4-capability;
  description
    "Identity for identification condition capability";
  reference
    "RFC 791: Internet Protocol - Identification";
}

identity ipv4-fragment-flags {
  base ipv4-capability;
  description
    "Identity for IPv4 fragment flags condition capability";
  reference
    "RFC 791: Internet Protocol - Fragmentation Flags";
}

identity exact-ipv4-fragment-offset {
  base ipv4-capability;
  description
    "Identity for exact-match IPv4 fragment offset condition capability";
  reference
    "RFC 791: Internet Protocol - Fragmentation Offset";
}
identity range-ipv4-fragment-offset {
  base ipv4-capability;
  description "Identity for range-match IPv4 fragment offset condition capability";
  reference "RFC 791: Internet Protocol - Fragmentation Offset";
}

identity exact-ipv4-ttl {
  base ipv4-capability;
  description "Identity for exact-match IPv4 Time-To-Live (TTL) condition capability";
  reference "RFC 791: Internet Protocol - Time To Live (TTL)";
}

identity range-ipv4-ttl {
  base ipv4-capability;
  description "Identity for range-match IPv4 Time-To-Live (TTL) condition capability";
  reference "RFC 791: Internet Protocol - Time To Live (TTL)";
}

identity ipv4-protocol {
  base ipv4-capability;
  description "Identity for IPv4 protocol condition capability";
  reference "RFC 790: Assigned numbers - Assigned Internet Protocol Number"
       RFC 791: Internet Protocol - Protocol";
}

identity exact-ipv4-address {
  base ipv4-capability;
  description "Identity for exact-match IPv4 address condition capability";
  reference "RFC 791: Internet Protocol - Address";
}

identity range-ipv4-address {
  base ipv4-capability;
}
description
"Identity for range-match IPv4 address condition capability";
reference
"RFC 791: Internet Protocol - Address";
}

identity ipv4-ip-opts {  
    base ipv4-capability;
    description
    "Identity for IPv4 option condition capability";
    reference
    "RFC 791: Internet Protocol - Options";
}

identity ipv4-geo-ip {  
    base ipv4-capability;
    description
    "Identity for geography condition capability";
    reference
    "draft-ietf-i2nsf-capability-04: Information Model of NSFs Capabilities - Geo-IP";
}

identity ipv6-capability {  
    base condition;
    description
    "Identity for IPv6 condition capabilities";
    reference
}

identity ipv6-traffic-class {  
    base ipv6-capability;
    description
    "Identity for IPv6 traffic class condition capability";
    reference
}

identity exact-ipv6-flow-label {  
    base ipv6-capability;
    description
    "Identity for exact-match IPv6 flow label condition capability";
}
identity range-ipv6-flow-label {
  base ipv6-capability;
  description
    "Identity for range-match IPv6 flow label condition capability";
  reference
}

identity exact-ipv6-payload-length {
  base ipv6-capability;
  description
    "Identity for exact-match IPv6 payload length condition capability";
  reference
    "RFC 8200: Internet Protocol, Version 6 (IPv6) Specification - Payload Length";
}

identity range-ipv6-payload-length {
  base ipv6-capability;
  description
    "Identity for range-match IPv6 payload length condition capability";
  reference
    "RFC 8200: Internet Protocol, Version 6 (IPv6) Specification - Payload Length";
}

identity ipv6-next-header {
  base ipv6-capability;
  description
    "Identity for IPv6 next header condition capability";
  reference
}

identity exact-ipv6-hop-limit {
  base ipv6-capability;
  description
    "Identity for exact-match IPv6 hop limit";
identity range-ipv6-capability {
  base ipv6-capability;
  description "Identity for range-match IPv6 capability";
}

identity range-ipv6-hop-limit {
  base ipv6-capability;
  description "Identity for range-match IPv6 hop limit condition capability";
}

identity exact-ipv6-address {
  base ipv6-capability;
  description "Identity for exact-match IPv6 address condition capability";
}

identity range-ipv6-address {
  base ipv6-capability;
  description "Identity for range-match IPv6 address condition capability";
}

identity tcp-capability {
  base condition;
  description "Identity for TCP condition capabilities";
  reference "RFC 793: Transmission Control Protocol";
}

identity exact-tcp-port-num {
  base tcp-capability;
  description "Identity for exact-match TCP port number"
identity range-tcp-port-num {
    base tcp-capability;
    description
        "Identity for range-match TCP port number condition capability";
    reference
        "RFC 793: Transmission Control Protocol - Port Number";
}

identity exact-tcp-seq-num {
    base tcp-capability;
    description
        "Identity for exact-match TCP sequence number condition capability";
    reference
        "RFC 793: Transmission Control Protocol - Sequence Number";
}

identity range-tcp-seq-num {
    base tcp-capability;
    description
        "Identity for range-match TCP sequence number condition capability";
    reference
        "RFC 793: Transmission Control Protocol - Sequence Number";
}

identity exact-tcp-ack-num {
    base tcp-capability;
    description
        "Identity for exact-match TCP acknowledgement number condition capability";
    reference
        "RFC 793: Transmission Control Protocol - Acknowledgement Number";
}

identity range-tcp-ack-num {
    base tcp-capability;
    description
        "Identity for range-match TCP acknowledgement number condition capability";
    reference
        "RFC 793: Transmission Control Protocol - Acknowledgement Number";
identity exact-tcp-window-size {
    base tcp-capability;
    description
        "Identity for exact-match TCP window size condition capability";
    reference
        "RFC 793: Transmission Control Protocol - Window Size";
}

identity range-tcp-window-size {
    base tcp-capability;
    description
        "Identity for range-match TCP window size condition capability";
    reference
        "RFC 793: Transmission Control Protocol - Window Size";
}

identity tcp-flags {
    base tcp-capability;
    description
        "Identity for TCP flags condition capability";
    reference
        "RFC 793: Transmission Control Protocol - Flags";
}

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

identity exact-udp-port-num {
    base udp-capability;
    description
        "Identity for exact-match UDP port number condition capability";
    reference
        "RFC 768: User Datagram Protocol - Port Number";
}

identity range-udp-port-num {
    base udp-capability;
    description
"Identity for range-match UDP port number condition capability";
reference
"RFC 768: User Datagram Protocol - Port Number";

identity exact-udp-total-length {
  base udp-capability;
  description
    "Identity for exact-match UDP total-length condition capability";
  reference
    "RFC 768: User Datagram Protocol - Total Length";
}

identity range-udp-total-length {
  base udp-capability;
  description
    "Identity for range-match UDP total-length condition capability";
  reference
    "RFC 768: User Datagram Protocol - Total Length";
}

identity icmp-capability {
  base condition;
  description
    "Identity for ICMP condition capabilities";
  reference
    "RFC 792: Internet Control Message Protocol";
}

identity icmp-type {
  base icmp-capability;
  description
    "Identity for ICMP type condition capability";
  reference
    "RFC 792: Internet Control Message Protocol";
}

identity url-capability {
  base condition;
  description
    "Identity for URL condition capabilities";
}

identity pre-defined {
  base url-capability;
}
description
    "Identity for URL pre-defined condition capabilities";
}

identity user-defined {
    base url-capability;
    description
        "Identity for URL user-defined condition capabilities";
}

identity log-action-capability {
    description
        "Identity for log-action capabilities";
}

identity rule-log {
    base log-action-capability;
    description
        "Identity for rule log log-action capability";
}

identity session-log {
    base log-action-capability;
    description
        "Identity for session log log-action capability";
}

identity ingress-action-capability {
    description
        "Identity for ingress-action capabilities";
    reference
        "draft-ietf-i2nsf-capability-04: Information Model of NSFs Capabilities - Action";
}

identity egress-action-capability {
    description
        "Base identity for egress-action capabilities";
    reference
        "draft-ietf-i2nsf-capability-04: Information Model of NSFs Capabilities - Egress action";
}

identity default-action-capability {
    description
        "Identity for default-action capabilities";
    reference
        "draft-ietf-i2nsf-capability-04: Information Model of NSFs Capabilities - Default action";
}
of NSFs Capabilities - Default action};
}

identity drop {
  base ingress-action-capability;
  base egress-action-capability;
  base default-action-capability;
  description
    "Identity for drop action capability";
  reference
    "draft-ietf-i2nsf-capability-04: Information Model
      of NSFs Capabilities - Actions and default action";
}

identity reject {
  base ingress-action-capability;
  base egress-action-capability;
  base default-action-capability;
  description
    "Identity for reject action capability";
  reference
    "draft-ietf-i2nsf-capability-04: Information Model
      of NSFs Capabilities - Actions and default action";
}

identity alert {
  base ingress-action-capability;
  base egress-action-capability;
  base default-action-capability;
  description
    "Identity for alert action capability";
  reference
    "draft-ietf-i2nsf-capability-04: Information Model
      of NSFs Capabilities - Actions and default action";
}

identity alert {
default action; }

identity mirror {
    base ingress-action-capability;
    base egress-action-capability;
    base default-action-capability;
    description
        "Identity for mirror action capability";
    reference
        "draft-ietf-i2nsf-capability-04: Information Model
         of NSFs Capabilities - Actions and
default action";
}

identity invoke-signaling {
    base egress-action-capability;
    description
        "Identity for invoke signaling action capability";
}

identity tunnel-encapsulation {
    base egress-action-capability;
    description
        "Identity for tunnel encapsulation action capability";
}

identity forwarding {
    base egress-action-capability;
    description
        "Identity for forwarding action capability";
}

identity redirection {
    base egress-action-capability;
    description
        "Identity for redirection action capability";
}

identity resolution-strategy-capability {
    description
        "Base identity for resolution strategy capability";
    reference
        "draft-ietf-i2nsf-capability-04: Information Model
         of NSFs Capabilities - Resolution Strategy";
}

identity fmr {

}
base resolution-strategy-capability;
description
"Identity for First Matching Rule (FMR)
resolution strategy capability";
reference
"draft-ietf-i2nsf-capability-04: Information Model
of NSFs Capabilities - Resolution Strategy";
}

identity lmr {
  base resolution-strategy-capability;
description
  "Identity for Last Matching Rule (LMR)
resolution strategy capability";
reference
  "draft-ietf-i2nsf-capability-04: Information Model
  of NSFs Capabilities - Resolution Strategy";
}

identity pmr {
  base resolution-strategy-capability;
description
    "Identity for Prioritized Matching Rule (PMR)
    resolution strategy capability";
reference
    "draft-ietf-i2nsf-capability-04: Information Model
    of NSFs Capabilities - Resolution Strategy";
}

identity pmre {
  base resolution-strategy-capability;
description
    "Identity for Prioritized Matching Rule
    with Errors (PMRE) resolution strategy capability";
reference
    "draft-ietf-i2nsf-capability-04: Information Model
    of NSFs Capabilities - Resolution Strategy";
}

identity pmrn {
  base resolution-strategy-capability;
description
    "Identity for Prioritized Matching Rule
    with No Errors (PMRN) resolution strategy capability";
reference
    "draft-ietf-i2nsf-capability-04: Information Model
    of NSFs Capabilities - Resolution Strategy";
}
identity advanced-nsf-capability {
  description
  "Base identity for advanced network security function (NSF) capabilities";
  reference
  "RFC 8329: Framework for Interface to Network Security Functions - Differences from ACL Data Models
draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller";
}

identity anti-virus-capability {
  base advanced-nsf-capability;
  description
  "Identity for advanced NSF anti-virus capabilities";
  reference
  "RFC 8329: Framework for Interface to Network Security Functions - Differences from ACL Data Models
draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller - Anti-virus";
}

identity anti-ddos-capability {
  base advanced-nsf-capability;
  description
  "Identity for advanced NSF anti-ddos capabilities";
  reference
  "RFC 8329: Framework for Interface to Network Security Functions - Differences from ACL Data Models
draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller - Anti-ddos";
}

identity ips-capability {
  base advanced-nsf-capability;
  description
  "Identity for advanced NSF Intrusion Prevention System (IPS) capabilities";
  reference
  "RFC 8329: Framework for Interface to Network Security Functions - Differences from ACL Data Models
draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller - Intrusion Prevention System";
}
identity voip-volte-capability {
    base advanced-nsf-capability;
    description "Identity for advanced NSF VoIP/VoLTE capabilities";
    reference "RFC 3261: SIP: Session Initiation Protocol
RFC 8329: Framework for Interface to Network Security Functions - Differences from ACL Data Models
draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller";
}

identity detect {
    base anti-virus-capability;
    description "Identity for advanced NSF anti-virus detect capability";
    reference "draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller - Anti-virus";
}

identity exception-application {
    base anti-virus-capability;
    description "Identity for advanced NSF anti-virus exception application capability";
    reference "draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller - Anti-virus";
}

identity exception-signature {
    base anti-virus-capability;
    description "Identity for advanced NSF anti-virus exception signature capability";
    reference "draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller - Anti-virus";
}

identity whitelists {
    base anti-virus-capability;
    description
identity syn-flood-action {
    base anti-ddos-capability;
    description
        "Identity for advanced NSF anti-DDoS syn flood action capability";
    reference
        "draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller - Anti-DDoS";
}

identity udp-flood-action {
    base anti-ddos-capability;
    description
        "Identity for advanced NSF anti-DDoS UDP flood action capability";
    reference
        "draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller - Anti-DDoS";
}

identity http-flood-action {
    base anti-ddos-capability;
    description
        "Identity for advanced NSF anti-DDoS http flood action capability";
    reference
        "draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller - Anti-DDoS";
}

identity https-flood-action {
    base anti-ddos-capability;
    description
        "Identity for advanced NSF anti-DDoS https flood action capability";
    reference
        "draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller - Anti-DDoS";
}
Advanced Security Functions with I2NSF Security Controller - Anti-DDoS;

identity dns-request-flood-action {
  base anti-ddos-capability;
  description
    "Identity for advanced NSF anti-DDoS dns request flood action capability";
  reference
    "draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller - Anti-DDoS";
}

identity dns-reply-flood-action {
  base anti-ddos-capability;
  description
    "Identity for advanced NSF anti-DDoS dns reply flood action capability";
  reference
    "draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller - Anti-DDoS";
}

identity icmp-flood-action {
  base anti-ddos-capability;
  description
    "Identity for advanced NSF anti-DDoS icmp flood action capability";
  reference
    "draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller - Anti-DDoS";
}

identity sip-flood-action {
  base anti-ddos-capability;
  description
    "Identity for advanced NSF anti-DDoS sip flood action capability";
  reference
    "draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller - Anti-DDoS";
}
identity detect-mode {
    base anti-ddos-capability;
    description
        "Identity for advanced NSF anti-DDoS detect mode capability";
    reference
        "draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller - Anti-DDoS";
}

identity baseline-learning {
    base anti-ddos-capability;
    description
        "Identity for advanced NSF anti-DDoS baseline learning capability";
    reference
        "draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller - Anti-DDoS";
}

identity signature-set {
    base ips-capability;
    description
        "Identity for advanced NSF IPS signature set capability";
    reference
        "draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller - Intrusion Prevention System";
}

identity ips-exception-signature {
    base ips-capability;
    description
        "Identity for advanced NSF IPS exception signature capability";
    reference
        "draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller - Intrusion Prevention System";
}

identity voice-id {
    base voip-volte-capability;
    description
        "Identity for advanced NSF VoIP/VoLTE voice-id
capability";
reference
   "RFC 3261: SIP: Session Initiation Protocol";
}

identity user-agent {
   base voip-volte-capability;
description
   "Identity for advanced NSF VoIP/VoLTE user agent
capability";
reference
   "RFC 3261: SIP: Session Initiation Protocol";
}

identity ipsec-capability {
   description
   "Base identity for an IPsec capabilities";
}

identity ike {
   base ipsec-capability;
description
   "Identity for an IPSec Internet Key Exchange (IKE)
capability";
}

identity ikeless {
   base ipsec-capability;
description
   "Identity for an IPSec without Internet Key Exchange (IKE)
capability";
}

/*
*  Grouping
*/

grouping nsf-capabilities {
   description
   "Network Security Funtion (NSF) Capabilities";
reference
   "RFC 8329: Framework for Interface to Network Security
Functions - I2NSF Flow Security Policy Structure
draft-ietf-i2nsf-capability-04: Information Model
of NSFs Capabilities - Capability Information Model Design";
leaf-list time-capabilities {
  type enumeration {
    enum absolute-time {
      description "absolute time capabilities.
      If network security function has the absolute time
      capability, the network security function supports rule execution according to absolute time."
    }
    enum periodic-time {
      description "periodic time capabilities.
      If network security function has the periodic time
      capability, the network security function supports rule execution according to periodic time."
    }
  }
  description "Time capabilities";
}

container event-capabilities {
  description "Capabilities of events.
  If network security function has
  the event capabilities, the network security functions
  supports rule execution according to system event
  and system alarm."

  reference "RFC 8329: Framework for Interface to Network Security
  Functions - I2NSF Flow Security Policy Structure
draft-ietf-i2nsf-capability-04: Information Model
of NSFs Capabilities - Design Principles and ECA
Policy Model Overview
draft-ietf-i2nsf-nsf-monitoring-data-model-01: A YANG
Data Model for Monitoring I2NSF Network Security
Functions - System Alarm and System Events";

  leaf-list system-event-capability {
    type identityref {
      base system-event-capability;
    }
    description "System event capabilities";
  }

  leaf-list system-alarm-capability {

type identityref {
  base system-alarm-capability;
}  
description
  "System alarm Capabilities";
}

container condition-capabilities {
  description
    "Conditions capabilities."
}

container generic-nsf-capabilities {
  description
    "Conditions capabilities.
    If a network security function has
    the condition capabilities, the network security function
    supports rule execution according to conditions of IPv4,
    IPv6, TCP, UDP, ICMP, and payload.";
  reference
    "RFC  791: Internet Protocol
RFC  792: Internet Control Message Protocol
RFC  793: Transmission Control Protocol
RFC 8200: Internet Protocol, Version 6 (IPv6)
  Specification - Next Header
RFC 8329: Framework for Interface to Network Security
Functions - I2NSF Flow Security Policy Structure
draft-ietf-i2nsf-capability-04: Information Model
of NSFs Capabilities - Design Principles and ECA Policy
Model Overview";
}

leaf-list ipv4-capability {
  type identityref {
    base ipv4-capability;
  }
  description
    "IPv4 packet capabilities";
  reference
    "RFC 791: Internet Protocol";
}

leaf-list ipv6-capability {
  type identityref {
    base ipv6-capability;
  }
  description
    "IPv6 packet capabilities";
  reference
    "RFC 791: Internet Protocol";
leaf-list tcp-capability {
  type identityref {
    base tcp-capability;
  }
  description
  "TCP packet capabilities";
  reference
  "RFC 793: Transmission Control Protocol";
}

leaf-list udp-capability {
  type identityref {
    base udp-capability;
  }
  description
  "UDP packet capabilities";
  reference
  "RFC 768: User Datagram Protocol";
}

leaf-list icmp-capability {
  type identityref {
    base icmp-capability;
  }
  description
  "ICMP packet capabilities";
  reference
  "RFC 8200: Internet Protocol, Version 6 (IPv6)";
}

container advanced-nsf-capabilities {
  description
  "Advanced Network Security Function (NSF) capabilities, such as anti-virus, anti-DDoS, IPS, and VoIP/VoLTE.";
  reference
  "RFC 8329: Framework for Interface to Network Security Functions - Differences from ACL Data Models
draft-dong-i2nsf-asf-config-01: Configuration of Advanced Security Functions with I2NSF Security Controller";

  leaf-list anti-virus-capability {
    type identityref {

base anti-virus-capability;
}
description
"Anti-virus capabilities";
reference
"draft-dong-i2nsf-asf-config-01: Configuration of
Advanced Security Functions with I2NSF Security
Controller";
}

leaf-list anti-ddos-capability {
  type identityref {
    base anti-ddos-capability;
  }
description
"Anti-ddos capabilities";
reference
"draft-dong-i2nsf-asf-config-01: Configuration of
Advanced Security Functions with I2NSF Security
Controller";
}

leaf-list ips-capability {
  type identityref {
    base ips-capability;
  }
description
"Intrusion Prevention System (IPS) capabilities";
reference
"draft-dong-i2nsf-asf-config-01: Configuration of
Advanced Security Functions with I2NSF Security
Controller";
}

leaf-list url-capability {
  type identityref {
    base url-capability;
  }
description
"URL capabilities";
reference
"draft-dong-i2nsf-asf-config-01: Configuration of
Advanced Security Functions with I2NSF Security
Controller";
}

leaf-list voip-volte-capability {
  type identityref {

base voip-volte-capability;
}
description
"VoIP and VoLTE capabilities";
reference
"draft-dong-i2nsf-asf-config-01: Configuration of
Advanced Security Functions with I2NSF Security
Controller";
}

leaf-list context-capabilities {
  type identityref {
    base context-capability;
  }
  description
    "Security context capabilities";
}

container action-capabilities {
  description
    "Action capabilities. If network security function has
    the action capabilities, it supports the attendant actions for policy rules.";

  leaf-list ingress-action-capability {
    type identityref {
      base ingress-action-capability;
    }
    description
      "Ingress-action capabilities";
  }

  leaf-list egress-action-capability {
    type identityref {
      base egress-action-capability;
    }
    description
      "Egress-action capabilities";
  }

  leaf-list log-action-capability {
    type identityref {
      base log-action-capability;
    }
    description

"Log-action capabilities";
}
}

leaf-list resolution-strategy-capabilities {
  type identityref {
    base resolution-strategy-capability;
  }
  description
    "Resolution strategy capabilities. The resolution strategies can be used to specify how to resolve conflicts that occur between the actions of the same or different policy rules that are matched for the same packet and by particular NSF";
  reference
    "draft-ietf-i2nsf-capability-04: Information Model of NSFs Capabilities - Resolution strategy";
}

leaf-list default-action-capabilities {
  type identityref {
    base default-action-capability;
  }
  description
    "Default action capabilities. A default action is used to execute I2NSF policy rules when no rule matches a packet. The default action is defined as pass, drop, reject, alert, or mirror.";
  reference
    "draft-ietf-i2nsf-capability-04: Information Model of NSFs Capabilities - Default action";
}

leaf-list ipsec-method {
  type identityref {
    base ipsec-capability;
  }
  description
    "IPsec method capabilities";
  reference
    "draft-ietf-i2nsf-sdn-ipsec-flow-protection-04";
}


/*
 * Data nodes
 */
7. IANA Considerations

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


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].

name: ietf-i2nsf-capability


prefix: nsfcap

reference: RFC XXXX

8. 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 transport secure transport is Secure Shell (SSH) [RFC6242]. The lowest RESTCONF layer
is HTTPS, and the required transport secure transport is TLS [RFC8446].

The NETCONF access control model [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 content.

There are a number of data nodes defined in this YANG module that are writable/creatable/deletable (i.e., config true, which is the default). These data nodes may be considered sensitive or vulnerable in some network environments. Write operations (e.g., edit-config) to these data nodes without proper protection can have a negative effect on network operations. These are the subtrees and data nodes and their sensitivity/vulnerability:

- ietf-i2nsf-capability: The attacker may provide incorrect information of the security capability of any target NSF by illegally modifying this.

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 and their sensitivity/vulnerability:

- ietf-i2nsf-capability: The attacker may gather the security capability information of any target NSF and misuse the information for subsequent attacks.

9. References

9.1. Normative References


9.2. Informative References

[draft-dong-i2nsf-asf-config]

[draft-ietf-i2nsf-capability]

[draft-ietf-i2nsf-nsf-facing-interface-dm]
[draft-ietf-i2nsf-nsf-monitoring-data-model]  

[draft-ietf-i2nsf-sdn-ipsec-flow-protection]  

[draft-ietf-i2nsf-terminology]  

[draft-ietf-supap-generic-policy-info-model]  
Appendix A. Configuration Examples

This section shows configuration examples of "ietf-i2nsf-capability" module for capabilities registration of general firewall.

A.1. Example 1: Registration for Capabilities of General Firewall

This section shows a configuration example for capabilities registration of general firewall.

```xml
<nsf xmlns="urn:ietf:params:xml:ns:yang:ietf-i2nsf-capability">
  <nsf-name>general_firewall</nsf-name>
  <condition-capabilities>
    <generic-nsf-capabilities>
      <ipv4-capability>ipv4-protocol</ipv4-capability>
      <ipv4-capability>exact-ipv4-address</ipv4-capability>
      <ipv4-capability>range-ipv4-address</ipv4-capability>
      <tcp-capability>exact-fourth-layer-port-num</tcp-capability>
      <tcp-capability>range-fourth-layer-port-num</tcp-capability>
    </generic-nsf-capabilities>
  </condition-capabilities>
  <action-capabilities>
    <ingress-action-capability>pass</ingress-action-capability>
    <ingress-action-capability>drop</ingress-action-capability>
    <ingress-action-capability>alert</ingress-action-capability>
    <egress-action-capability>pass</egress-action-capability>
    <egress-action-capability>drop</egress-action-capability>
    <egress-action-capability>alert</egress-action-capability>
  </action-capabilities>
</nsf>
```

Figure 4: Configuration XML for Capabilities Registration of General Firewall

Figure 4 shows the configuration XML for capabilities registration of general firewall and its capabilities are as follows.

1. The name of the NSF is general_firewall.
2. The NSF can inspect protocol, exact IPv4 address, and range IPv4 address for IPv4 packets.
3. The NSF can inspect exact port number and range port number for fourth layer packets.
4. The NSF can control whether the packets are allowed to pass, drop, or alert.

A.2. Example 2: Registration for Capabilities of Time based Firewall

This section shows a configuration example for capabilities registration of time based firewall.

<nsf xmlns="urn:ietf:params:xml:ns:yang:ietf-i2nsf-capability">
  <nsf-name>time_based_firewall</nsf-name>
  <time-capabilities>absolute-time</time-capabilities>
  <time-capabilities>periodic-time</time-capabilities>
  <condition-capabilities>
    <generic-nsf-capabilities>
      <ipv4-capability>ipv4-protocol</ipv4-capability>
      <ipv4-capability>exact-ipv4-address</ipv4-capability>
      <ipv4-capability>range-ipv4-address</ipv4-capability>
    </generic-nsf-capabilities>
  </condition-capabilities>
  <action-capabilities>
    <ingress-action-capability>pass</ingress-action-capability>
    <ingress-action-capability>drop</ingress-action-capability>
    <ingress-action-capability>alert</ingress-action-capability>
    <egress-action-capability>pass</egress-action-capability>
    <egress-action-capability>drop</egress-action-capability>
    <egress-action-capability>alert</egress-action-capability>
  </action-capabilities>
</nsf>

Figure 5: Configuration XML for Capabilities Registration of Time based Firewall

Figure 5 shows the configuration XML for capabilities registration of time based firewall and its capabilities are as follows.

1. The name of the NSF is time_based_firewall.
2. The NSF can execute the security policy rule according to absolute time and periodic time.
3. The NSF can inspect protocol, exact IPv4 address, and range IPv4 address for IPv4 packets.
4. The NSF can control whether the packets are allowed to pass, drop, or alert.
A.3. Example 3: Registration for Capabilities of Web Filter

This section shows a configuration example for capabilities registration of web filter.

```xml
<nsf xmlns="urn:ietf:params:xml:ns:yang:ietf-i2nsf-capability">
  <nsf-name>web_filter</nsf-name>
  <condition-capabilities>
    <advanced-nsf-capabilities>
      <url-capability>user-defined</url-capability>
    </advanced-nsf-capabilities>
  </condition-capabilities>
  <action-capabilities>
    <ingress-action-capability>pass</ingress-action-capability>
    <ingress-action-capability>drop</ingress-action-capability>
    <ingress-action-capability>alert</ingress-action-capability>
    <egress-action-capability>pass</egress-action-capability>
    <egress-action-capability>drop</egress-action-capability>
    <egress-action-capability>alert</egress-action-capability>
  </action-capabilities>
</nsf>
```

Figure 6: Configuration XML for Capabilities Registration of Web Filter

Figure 6 shows the configuration XML for capabilities registration of web filter and its capabilities are as follows.

1. The name of the NSF is web_filter.
2. The NSF can inspect url for http and https packets.
3. The NSF can control whether the packets are allowed to pass, drop, or alert.

A.4. Example 4: Registration for Capabilities of VoIP/VoLTE Filter

This section shows a configuration example for capabilities registration of VoIP/VoLTE filter.
Figure 7 shows the configuration XML for capabilities registration of VoIP/VoLTE filter and its capabilities are as follows.

1. The name of the NSF is voip_volte_filter.
2. The NSF can inspect voice id for VoIP/VoLTE packets.
3. The NSF can control whether the packets are allowed to pass, drop, or alert.

### A.5. Example 5: Registration for Capabilities of HTTP and HTTPS Flood Mitigation

This section shows a configuration example for capabilities registration of http and https flood mitigation.
<nsf xmlns="urn:ietf:params:xml:ns:yang:ietf-i2nsf-capability">
  <nsf-name>http_and_https_flood_mitigation</nsf-name>
  <condition-capabilities>
    <advanced-nsf-capabilities>
      <anti-ddos-capability>http-flood-action</anti-ddos-capability>
      <anti-ddos-capability>https-flood-action</anti-ddos-capability>
    </advanced-nsf-capabilities>
  </condition-capabilities>
  <action-capabilities>
    <ingress-action-capability>pass</ingress-action-capability>
    <ingress-action-capability>drop</ingress-action-capability>
    <ingress-action-capability>alert</ingress-action-capability>
    <egress-action-capability>pass</egress-action-capability>
    <egress-action-capability>drop</egress-action-capability>
    <egress-action-capability>alert</egress-action-capability>
  </action-capabilities>
</nsf>

Figure 8: Configuration XML for Capabilities Registration of HTTP and HTTPS Flood Mitigation

Figure 8 shows the configuration XML for capabilities registration of http and https flood mitigation and its capabilities are as follows.

1. The name of the NSF is http_and_https_flood_mitigation.
2. The location of the NSF is 221.159.112.140.
3. The NSF can control the amount of packets for http and https packets.
4. The NSF can control whether the packets are allowed to pass, drop, or alert.

Appendix B. Changes from draft-ietf-i2nsf-capability-data-model-04

The following changes are made from draft-ietf-i2nsf-capability-data-model-04:

- The version is revised according to the comments from Acee Lindem and Carl Moberg who are YANG doctors for review.

Appendix C. 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).

Appendix D. Contributors

This document is made by the group effort of I2NSF working group. Many people actively contributed to this document. The following are considered co-authors:

- Hyoungshick Kim (Sungkyunkwan University)
- Daeyoung Hyun (Sungkyunkwan University)
- Dongjin Hong (Sungkyunkwan University)
- Liang Xia (Huawei)
- Jung-Soo Park (ETRI)
- Tae-Jin Ahn (Korea Telecom)
- Se-Hui Lee (Korea Telecom)

Authors’ Addresses

Susan Hares
Huawei
7453 Hickory Hill
Saline, MI  48176
USA

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

Jaehoon Paul Jeong
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
Fax: +82 31 290 7996
EMail: pauljeong@skku.edu
URI: http://iotlab.skku.edu/people-jaehoon-jeong.php