Revision Letter

I2NSF Network Security Function-Facing Interface YANG Data Model

- **Old Draft Name: draft-ietf-i2nsf-nsf-facing-interface-dm-06**
- **New Draft Name: draft-ietf-i2nsf-nsf-facing-interface-dm-07**

Jaehoon Paul Jeong

11/4/2019

Dear Acee Lindem,

I sincerely appreciate your valuable comments on the NSF-Facing Interface Data Model document. Your comments use a bold font and my answers use a regular font with the prefix [PAUL].

Document: draft-ietf-i2nsf-nsf-facing-interface-dm-06

Reviewer: Acee Lindem

Review Date: June 22, 2019

Review Type: Working Group Last Call

Intended Status: Standards Track

Summary: Needs to go back to Working Group for rework and another WGLC

Modules: "ietf-i2nsf-policy-rule-for-nsf@2019-06-12.yang"

Tech Summary: The model defines different types of I2NSF security policy. Each is comprised of an event, a condition, and an action. There is significant overlap with other IETF models. Within I2NSF, there is repetition of definitions which needs to go into a common I2NSF types module. Additionally, the data descriptions were done quickly and never reviewed or edited. I believe it needs to go back to the working group for another revision and working group last call.

Major Comments:

1. Why dont you leverage the definitions in RFC 8519 for packet matching? We dont need all this defined again.

=> [PAUL] We tried to refer to RFC 8519 as your opinion. However, in the I2NSF framework, NSF Capabilities are first registered to Security Controller according to Capability YANG data model. Then, when the security policy is delivered by the I2NSF User, the security policy is automatically matched with the security capabilities registered in the Security Controller, and the policy is delivered to the NSF in accordance with the NSF-Facing Interface YANG data model (Please refer https://datatracker.ietf.org/doc/draft-yang-i2nsf-security-policy-translation/).

Also, we are considering content security control or attack mitigation control as well as a simple ACL. So, we need a new ACL YANG data model.

NSF-Facing Interface YANG Data Model:

```
case range-match {
  list range-ipv4-header-length {
    key "start-ipv4-header-length
         end-ipv4-header-length";
    leaf start-ipv4-header-length {
      type uint8 {
        range "5..15";
      }
      description
        "Starting IPv4 header length for a range match.";
    }

    leaf end-ipv4-header-length {
      type uint8 {
        range "5..15";
      }
      description
        "Ending IPv4 header length for a range match.";
    }
    description
      "Range match for an IPv4 header length.";
  }
}
```

Capability YANG Data Model:

```
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";
}
```

2. Date and time are defined in RFC 6991. Why dont those suffice?

=> [PAUL] We revised the date-and-time according to your comments.

NEW:

```
   container start-time {
                        uses "key-chain:lifetime";
                        description
                           "Start time when the rules are applied";
                        reference
                           "RFC 8177: YANG Data Model for Key Chains
                                   - lifetime";
                     }
                     container end-time {
                        uses "key-chain:lifetime";
                        description
                           "End time when the rules are applied";
                        reference
                           "RFC 8177: YANG Data Model for Key Chains
                                   - lifetime";
                     }
```

3. Refer to the intervals as "time-intervals" rather than "time-zones". The term "time-zone" has a completely different connotation.

=> [PAUL] We changed the time-zones to time-intervals

OLD:

```
 |       |   +--rw time-zones
   |       |   |   +--rw absolute-time-zone
   |       |   |   |   +--rw start-time?     start-time-type
   |       |   |   |   +--rw end-time?       end-time-type
   |       |   |   +--rw periodic-time-zone
   |       |   |       +--rw day
   |       |   |       |   +--rw every-day?          boolean
```

```
|       |  |       |  +--rw specific-day*     day-type
|       |  |       +--rw month
|       |  |          +--rw every-month?          boolean
|       |  |          +--rw specific-month*     month-type
```

NEW:

```
|       |  +--rw time-intervals
|       |  |  +--rw absolute-time-interval
|       |  |  |  +--rw start-time?     start-time-type
|       |  |  |  +--rw end-time?       end-time-type
|       |  |  +--rw periodic-time-interval
|       |  |       +--rw day
|       |  |       |  +--rw every-day?          boolean
|       |  |       |  +--rw specific-day*     day-type
|       |  |       +--rw month
|       |  |          +--rw every-month?          boolean
|       |  |          +--rw specific-month*     month-type
```

4. What the "acl-number"? Also, ACLs are named (RFC 8519). Also, why define all the packet matching and then reference an ACL.

=> [PAUL] We delete acl-number. We tried to refer to RFC 8519 as your opinion. However, in the I2NSF framework, NSF Capabilities are first registered to Security Controller according to Capability YANG data model. Then, when the security policy is delivered by the I2NSF User, the security policy is automatically matched with the security capabilities registered in the Security Controller, and the policy is delivered to the NSF in accordance with the NSF-Facing Interface YANG data model (Please refer https://datatracker.ietf.org/doc/draft-yang-i2nsf-security-policy-translation/).

Also, we are considering content security control or attack mitigation control as well as a simple ACL. So, we need a new ACL YANG data model.

5. The descriptions are very awkwardly worded and in many cases simply repeat the data node or identify description without hyphens. I started trying to fix this but it was too much. Ill pass for on for some examples. There are enough co-authors and contributors that one would expect much better.

=> [PAUL] Thank you for trying to remedy the awkward words. I checked the sentences that you revise. Thank you for your efforts.

6. There is overlap of definitions with the I2NSF capabilities draft. The common types and identities should be factored into a common I2NSF types module.

=> [PAUL] There are lots of the same names in the NSF-Facing Interface and Capability YANG data models for automation, but the Capability YANG data model has a purpose for capability registration, but the NSF-Facing Interface YANG data model has a purpose for configuration. Therefore, identities with the same name have a different base for each data model as follows:

NSF-Facing Interface YANG Data Model

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

Capability YANG Data Model

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

If you want the bases of identities with the same name to be the same, please let us know to revise them.

7. The "Security Considerations" in section 8 do not conform to the recommended template in https://trac.ietf.org/trac/ops/wiki/yang-security-guidelines>

=> [PAUL] I revised “Security Considerations” Section according to the recommended template.

NEW:

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-policy-rule-for-nsf: The attacker may provide incorrect policy information of any target NSFs 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-policy-rule-for-nsf: The attacker may gather the security policy information of any target NSFs and misuse the security policy information for subsequent attacks.

Minor Comments:

1. Section 3.1 should reference RFC8340 rather than attempting to include tree diagram formatting semantics.

=> [PAUL] We revised Section 3.1 such that it references RFC 8340 instead of including tree diagram formatting semantics.

OLD:

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 [RFC8340] is as follows:

- Brackets "[" and "]" enclose list keys.
- Abbreviations before data node names: "rw" means configuration (read-write) and "ro" state data (read-only).
- Symbols after data node names: "?" means an optional node and "*" denotes a "list" and "leaf-list".
- Parentheses enclose choice and case nodes, and case nodes are also marked with a colon (":").

- Ellipsis ("...") stands for contents of subtrees that are not shown.

NEW:

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]

2. "iiprfn" is a poor choice for default model prefix - I suggest "nsfintf". It is only one character longer and actually is expands to something meaningful.

=> [PAUL] I agree at your comments on modifying “iipfn” into “nsfintf”.

3. RFC 2460 is obsoleted by RFC 8200.

=> [PAUL] We changed from RFC 2460 to RFC 8200 as a reference.

4. RFC 791 is the wrong reference for IPv4 TOS. It should be RFC 1394.

=> [PAUL] We changed from RFC 791 to RFC 1394 for the reference to IPv4 TOS.

5. What is the IGRP protocol? Im familiar with EIGRP but not IGRP.

=> [PAUL] IGRP stands for Interior Gateway Routing Protocol. Refer to https://www.cisco.com/c/en/us/support/docs/ip/interior-gateway-routing-protocol-igrp/26825-5.html.
We consider both IGRP protocol and EIGRP protocol in this document.

6. What is the skip protocol? Is this about skipping the check? If so, why is it needed.

=> [PAUL] Skip was one of the ICMP type. However, it is an unsupported ICMP type, so I removed it in this version.

7. Reference for IPv6 ICMP should be RFC 2463.

=> [PAUL] Since RFC 4443 obsoletes RFC 2463 for IPv6 ICMP (i.e., ICMPv6), this version refers to RFC 4443.

8. Why do you include Photuris definitions? Nobody uses this.

=> [PAUL] I delete the Photuris according to your comments.

9. Note that all the keys for all config true lists must be unique so your specification in the description as well as mandatory true are redundant for the rules list. This mistake is in other lists as well.

=> [PAUL] We deleted the redundant mandatory true according to your comments.

10. What is during time?

=> [PAUL] I replace “during” with “duration” since it represents the duration of a long-lived connection. Also, I revised the description for “leaf during” from “This has long-connection during a time” to “This is the duration of the long-connection”.

OLD:

```
leaf during {
        type uint16;
        description
          "This is during time.";
  }
```

NEW:

```
leaf duration {
        type uint16;
        description
          " This is the duration of the long-connection.";
  }
```

11. What is a "security-grup"? Is this a security-group?

=> [PAUL] I changed security-grup to security-group.

12. The module prologue doesnt match the example in Appendix B of RFC 8407.

=> [PAUL] I revised according to your comments in the module prologue.

13. There needs to be a good definition of absolute and periodic time in the descriptions.

=> [PAUL] We clarified the definitions of absolute time and periodic time in the descriptions as follows:

OLD:

absolute-time-zone: Rule execution according to absolute time.

periodic-time-zone: Rule execution according to periodic time.

NEW:

absolute-time-interval: Rule execution time according to the absolute time. The absolute time interval means the exact time to start or end.

periodic-time-interval: Rule execution time according to the periodic time. The periodic time interval means the repeated time such as a day, week, and month.

14. The References do not include all the RFCs referenced by YANG model reference statements.

=> [PAUL] We added RFCs referenced by YANG model reference statements such as RFC 768, RFC 3232, RFC 791, RFC 792, RFC 793, RFC 3261, RFC 4443, and RFC 8200.

Thanks.

Best Regards,

Paul

--

==========

Mr. Jaehoon (Paul) Jeong, Ph.D.

Associate Professor

Department of Software

Sungkyunkwan University

Office: +82-31-299-4957

Email: jaehoon.paul@gmail.com, pauljeong@skku.edu

Personal Homepage: http://iotlab.skku.edu/people-jaehoon-jeong.php