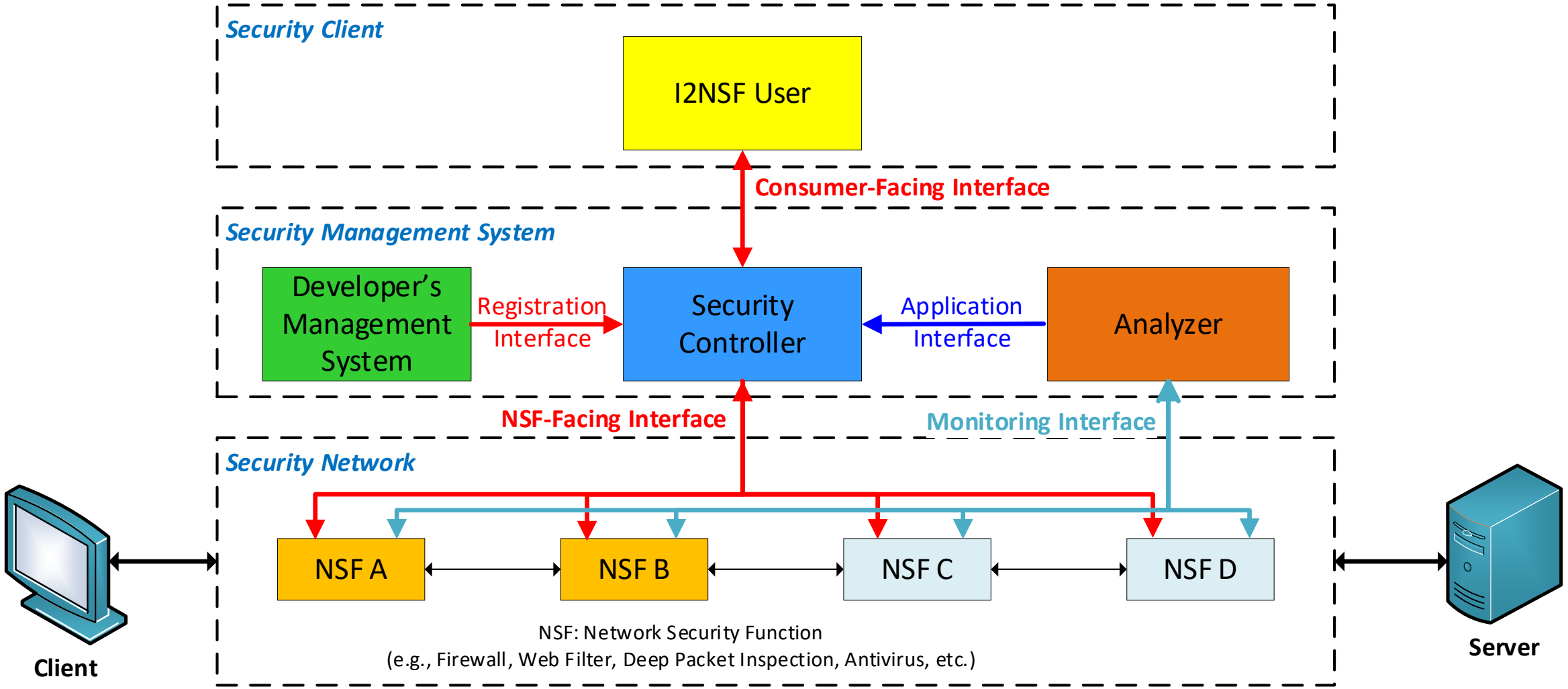
I2NSF YANG Data Model Comparison

draft-ietf-i2nsf-consumer-facing-interface-dm-17
draft-ietf-i2nsf-nsf-facing-interface-dm-22

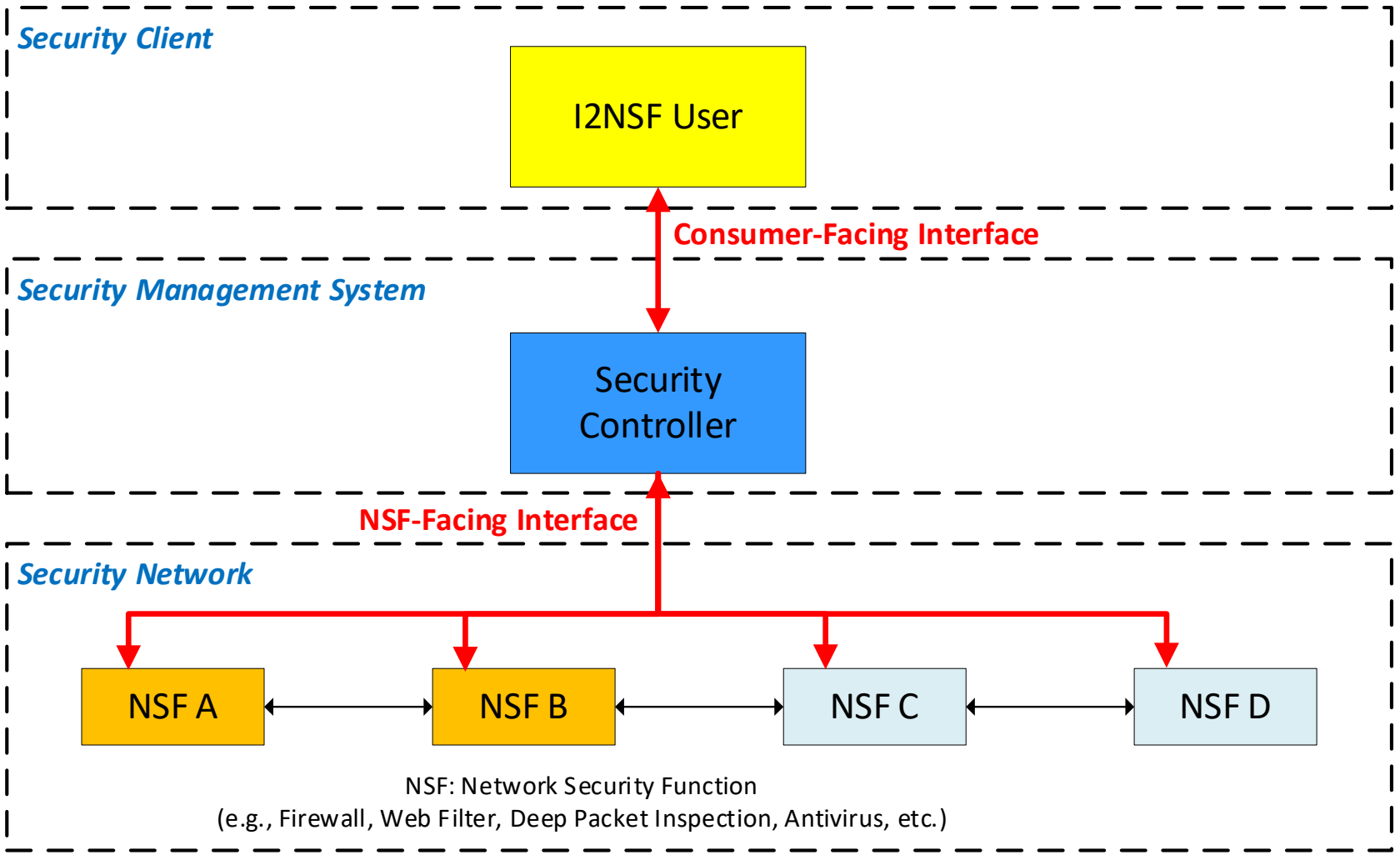
IETF 113, Vienna
March 24th, 2022

Jaehoon (Paul) Jeong and Patrick Lingga
{pauljeong, patricklink}@skku.edu
Sungkyunkwan University

I2NSF Framework



I2NSF Framework – Consumer-Facing Interface and NSF-Facing Interface



Objectives of Consumer-Facing Interface and NSF-Facing Interface (1/2)

- **Consumer-Facing Interface (CFI):**
  - It is assumed that vendors also provide front-end web applications to an I2NSF User.
  - The Consumer-Facing Interface is required because the web applications developed by each vendor need to have a standard interface specifying the data types used when the I2NSF User and Security Controller communicate with each other using this interface.
  - Therefore, Consumer-Facing Interface document specifies the required information, their data types, and encoding schemes so that high-level security policies (or configuration information for security policies) can be transferred to the Security Controller through the Consumer-Facing Interface.
  - These high-level policies can be translated into low-level security policies by the Security Controller.

Objectives of Consumer-Facing Interface and NSF-Facing Interface (2/2)

- **NSF-Facing Interface (NFI):**
  - The NSF-Facing Interface focuses on providing security policy configuration for the NSFs as a low-level policy that can be used by the NSFs to deploy security services.
  - The Security Controller delivers the translated low-level policies to Network Security Functions (NSFs) according to their respective security capabilities for the required security enforcement.
  - The data model provides Access Control Lists (ACLs), i.e., a generic NSF (operate on packet header for layer 2, layer3, and layer 4), and an advanced NSF (Intrusion Prevention System, URL-Filtering, anti-DDoS, Antivirus, and Voice over Internet Protocol (VoIP) or Voice over Cellular Network (VoCN) Filter).
  - The ACLs provided in the NSF-Facing Interface YANG data model is imported from RFC 8519 (YANG Data Model for Network Access Control Lists (ACLs)).

Top-Level YANG Tree Comparison

Consumer-Facing Interface (CFI):

```
module: ietf-i2nsf-cfi-policy
  +--rw i2nsf-cfi-policy* [name]
     +--rw name                    string
     +--rw language?               string
     +--rw resolution-strategy?    identityref
     +--rw rules* [name]
     |  ...
     +--rw endpoint-groups
     |  ...
     +--rw threat-prevention
        ...
```

NSF-Facing Interface (NFI):

```
module: ietf-i2nsf-policy-rule-for-nsf
  +--rw i2nsf-security-policy* [name]
     +--rw name                    string
     +--rw language?               string
     +--rw priority-usage?         identityref
     +--rw resolution-strategy?    identityref
     +--rw default-action?         identityref
     +--rw rules* [name]
     |  ...
     +--rw rule-group
        ...
```

- The top-level CFI and NFI YANG data model provide the language-tag and resolution-strategy.

- default action and priority usage are not provided in CFI YANG data model.
  - **Reason:** The Security Policy Translator can set these both default action and priority usage to the low-level security policy.
  - **Philosophy of CFI:** To make CFI as simple as possible.

- In CFI, endpoint groups and threat prevention are used to register information (e.g., mapping a user to an IP address) with the database for high-level configuration.
  - **endpoint groups:** user-group, device-group, location-group, and url-group
  - **threat prevention:** threat-feed-list and payload-content

Rule-Level YANG Tree Comparison

Consumer-Facing Interface (CFI):

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

NSF-Facing Interface (NFI):

```
 +--rw rules* [name]
     |  +--rw name                 string
     |  +--rw description?         string
     |  +--rw priority?            uint8
     |  +--rw enable?              boolean
     |  +--rw long-connection
     |  |  +--rw enable?      boolean
     |  |  +--rw duration?    uint32
     |  +--rw event
     |  |  ...
     |  +--rw condition
     |  |  ...
     |  +--rw action
     |     ...
```

- The CFI and NFI data model use the Event-Condition-Action (ECA) policy rule with priority for the rule is provided in both YANG data model.

- long-connection (i.e., a connection that is maintained after the socket connection is established) is provided in NFI to handle stateful network service.
  - **Reason:** The Security Policy Translator can set this long-connection to the low-level security policy.
  - **Philosophy of CFI:** To make CFI as simple as possible.

- The contents of the ECA is different for CFI and NFI data model as shown in the next slides.

Event YANG Tree Comparison

Consumer-Facing Interface (CFI):

```
|  +--rw event
|  |  +--rw system-event*   identityref
|  |  +--rw system-alarm*   identityref
```

NSF-Facing Interface (NFI):

```
|  +--rw event
|  |  +--rw description?     string
|  |  +--rw system-event*   identityref
|  |  +--rw system-alarm*   identityref
```

- CFI and NFI have the almost same structures for Event except for description in NFI.
  - description is optional because it contains human-readable text for the description of an event.
- System Event: Access Violation and Configuration Change
- System Alarm: Memory, CPU, Disk, Hardware, and Interface Alarm

Condition YANG Tree Comparison – Layers 2, 3, and 4 (1/2)

Consumer-Facing Interface (CFI):

```
+--rw condition
|  +--rw firewall
|  |  +--rw source*      union (user-group or device-group name)
|  |  +--rw destination* union (user-group or device-group name)
|  |  +--rw transport-layer-protocol?   identityref
|  |  +--rw range-port-number
|  |  |  +--rw start-port-number?   inet:port-number
|  |  |  +--rw end-port-number?     inet:port-number
|  |  |  +--rw icmp
|  |  |     +--rw message*   identityref

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

NSF-Facing Interface (NFI):

```
|  +--rw condition
|  |  +--rw description?    string
|  |  +--rw layer-2* [destination-mac-address source-mac-address
                      ethertype]
|  |  |  ...
|  |  +--rw (layer-3)?
|  |  |  +--:(ipv4)
|  |  |  |  ...
|  |  |  +--:(ipv6)
|  |  |     ...
|  |  +--rw (layer-4)?
|  |  |  +--:(tcp)
|  |  |  |  ...
|  |  |  +--:(udp)
|  |  |  |  ...
|  |  |  +--:(sctp)
|  |  |  |  ...
|  |  |  +--:(dccp)
|  |  |  |  ...
|  |  |  +--:(icmp)
|  |  |     ...
```

Condition YANG Tree Comparison – Layers 2, 3, and 4 (2/2)

- CFI aims at an easy security policy configuration.
  - CFI YANG data model provides a way to save IP addresses of user/device into a database to be used for easy configuration of ACLs.
  - Some elements in an NFI security policy are handled by Security Policy Translator.

- In CFI YANG data model, the firewall condition (Access Control Lists (ACLs)) consists of
  - Source and destination MAC addresses,
  - Source and destination IP (IPv4 or IPv6) addresses,
  - Type of transport protocol (i.e., TCP, UDP, SCTP, DCCP),
  - Source and destination port numbers,
  - Type of application protocol,
  - ICMP type and code (for ICMPv4 and ICMPv6).

- The NFI YANG data model provides more fields that cover most headers of the protocols (Based on RFC8519 (ACLs) – IP (IPv4 or IPv6)).
  - IPv4 covers DSCP (Differentiated Services Code Point), ECN (Explicit Congestion Notification), length (total length), ttl, protocol, IHL (Internet Header Length), flags, offset, identification, source addresses, and destination addresses fields.
  - IPv6 covers DSCP, ECN, length (Payload Length), ttl (Hop Limit), protocol (Next Header in IPv6), source addresses, and destination addresses fields.
  - TCP covers source ports, destination ports, sequence number, acknowledgement number, data-offset, reserved, flags, window-size, urgent-pointer, and options fields.
  - UDP covers source ports, destination ports, and length fields.
  - SCTP covers source ports, destination ports, chunk type, and chunk length fields.
  - DCCP covers source ports, destination ports, service code, type, and data offset fields.

Condition YANG Tree Comparison – Advanced NSFs: DDoS, Antivirus, Payload (DPI), URL Filtering, Voice Filtering (1/3)

Consumer-Facing Interface (CFI):

```
|  +--rw condition
|  |  ...
|  |  +--rw ddos
|  |  |  +--rw rate-limit
|  |  |     +--rw packet-rate-threshold?   uint64
|  |  |     +--rw byte-rate-threshold?     uint64
|  |  |     +--rw flow-rate-threshold?     uint64
|  |  +--rw anti-virus
|  |  |  +--rw exception-files*   string
|  |  +--rw payload
|  |  |  +--rw content*
   -> /i2nsf-cfi-policy/threat-prevention/payload-content/name
|  |  +--rw url-category
|  |  |  +--rw url-name?
            -> /i2nsf-cfi-policy/endpoint-groups/url-group/name
|  |  +--rw voice
|  |  |  +--rw source-id*        string
|  |  |  +--rw destination-id*   string
|  |  |  +--rw user-agent*       string
|  |  +--rw threat-feed
|  |     +--rw name*
   -> /i2nsf-cfi-policy/threat-prevention/threat-feed-list/name

+--rw threat-prevention
       +--rw threat-feed-list* [name]
       |  +--rw name           string
       |  +--rw description?   string
       |  +--rw signatures*    identityref
       +--rw payload-content* [name]
          +--rw name           string
          +--rw description?   string
          +--rw content*       binary

+--rw endpoint-groups
    |  +--rw url-group* [name]
    |     +--rw name    string
    |     +--rw url*    string
```

NSF-Facing Interface (NFI):

```
|  +--rw condition
|  |  ...
|  |  +--rw ddos
|  |  |  +--rw description?          string
|  |  |  +--rw alert-packet-rate?    uint32
|  |  |  +--rw alert-flow-rate?      uint32
|  |  |  +--rw alert-byte-rate?      uint32
|  |  +--rw anti-virus
|  |  |  +--rw profile*            string
|  |  |  +--rw exception-files*    string
|  |  +--rw payload
|  |  |  +--rw description?   string
|  |  |  +--rw content*       binary
|  |  +--rw url-category
|  |  |  +--rw description?    string
|  |  |  +--rw pre-defined*    string
|  |  |  +--rw user-defined*   string
|  |  +--rw voice
|  |  |  +--rw description?              string
|  |  |  +--rw source-voice-id*          string
|  |  |  +--rw destination-voice-id*     string
|  |  |  +--rw user-agent*               string
```

- **Note**
  - The registration of a pair of (name, value) for a condition in CFI needs to be done to Security Controller by CFI YANG data model.
  - With this, Security Policy Translator can perform a policy translation.

Condition YANG Tree Comparison – Advanced NSFs: DDoS, Antivirus, Payload (DPI), URL Filtering, Voice Filtering (2/3)

- The CFI and NFI YANG data models are similar for DDoS, Antivirus, Payload (DPI), URL Filtering, and Voice Filtering conditions.
- The difference is that in CFI some of the information (name, value) for configuration is saved into a database in Security Controller for easy configuration.
- The configuration can be done by using the key name that holds the corresponding value.
- The registration for the database can be done with the following Xpath (i.e., used to navigate through elements in an XML document):
  - /i2nsf-cfi-policy/threat-prevention/payload-content/name
  - /i2nsf-cfi-policy/endpoint-groups/url-group/name
  - /i2nsf-cfi-policy/threat-prevention/threat-feed-list/name

Condition YANG Tree Comparison – Advanced NSFs: DDoS, Antivirus, Payload (DPI), URL Filtering, Voice Filtering (3/3)

- XML Example of the registration for the database with XPath:

```
<i2nsf-cfi-policy
 xmlns="urn:ietf:params:xml:ns:yang:ietf-i2nsf-cfi-policy">
  <name>security_policy_for_blocking_sns</name>
  <endpoint-groups>
    <user-group>
      <name>employees</name>
      <range-ipv4-address>
        <start-ipv4-address>192.0.2.11</start-ipv4-address>
        <end-ipv4-address>192.0.2.90</end-ipv4-address>
      </range-ipv4-address>
    </user-group>
    <device-group>
      <name>webservers</name>
      <range-ipv4-address>
        <start-ipv4-address>198.51.100.11</start-ipv4-address>
        <end-ipv4-address>198.51.100.20</end-ipv4-address>
      </range-ipv4-address>
      <application-protocol>nsfcfi:http</application-protocol>
      <application-protocol>nsfcfi:https</application-protocol>
    </device-group>
    <url-group>
      <name>sns-websites</name>
      <url>example1.com</url>
      <url>example2.com</url>
    </url-group>
  </endpoint-groups>
</i2nsf-cfi-policy>
```

Condition YANG Tree Comparison – Context (1/3)

Consumer-Facing Interface (CFI):

```
|   |  +--rw context
|   |   |  +--rw time
|   |   |   |  +--rw start-date-time?   yang:date-and-time
|   |   |   |  +--rw end-date-time?     yang:date-and-time
|   |   |   |  +--rw period
|   |   |   |   |  +--rw start-time?   time
|   |   |   |   |  +--rw end-time?     time
|   |   |   |   |  +--rw day*          day
|   |   |   |   |  +--rw date*         int8
|   |   |   |   |  +--rw month*        string
|   |   |   |   +--rw frequency?         enumeration
|   |   |   +--rw application
|   |   |   |  +--rw protocol*      identityref
|   |   |   +--rw device-type
|   |   |   |  +--rw device*        identityref
|   |   |   +--rw users
|   |   |   |  +--rw user* [id]
|   |   |   |   |  +--rw id      uint32
|   |   |   |   |  +--rw name?   string
|   |   |   |  +--rw group* [id]
|   |   |   |     +--rw id      uint32
|   |   |   |     +--rw name?   string
```

NSF-Facing Interface (NFI):

```
|   |  +--rw context
|   |     +--rw description?           string
|   |     +--rw time
|   |      |  +--rw start-date-time?   yang:date-and-time
|   |      |  +--rw end-date-time?     yang:date-and-time
|   |      |  +--rw period
|   |      |   |  +--rw start-time?   time
|   |      |   |  +--rw end-time?     time
|   |      |   |  +--rw day*          day
|   |      |   |  +--rw date*         int8
|   |      |   |  +--rw month*        string
|   |      |   +--rw frequency?         enumeration
|   |      +--rw application
|   |      |  +--rw description?   string
|   |      |  +--rw protocol*      identityref
|   |      +--rw device-type
|   |      |  +--rw description?   string
|   |      |  +--rw device*        identityref
|   |      +--rw users
|   |      |  +--rw description?   string
|   |      |  +--rw user* [id]
|   |      |   |  +--rw id      uint32
|   |      |   |  +--rw name?   string
|   |      |  +--rw group* [id]
|   |      |     +--rw id      uint32
|   |      |     +--rw name?   string
```

- **Note**
  - context contains extra information for filtering.
  - The contents of context in CFI are the same with those of context in NFI except the element of “description” in NFI.

Condition YANG Tree Comparison – Context (2/3)

Consumer-Facing Interface (CFI):

```
|   |  +--rw context
|   |  |  ...
|   |  |  +--rw geographic-location
|   |  |     +--rw source*
        -> /i2nsf-cfi-policy/endpoint-groups/location-group/name
|   |  |     +--rw destination*
        -> /i2nsf-cfi-policy/endpoint-groups/location-group/name

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

NSF-Facing Interface (NFI):

```
|   |  +--rw context
|   |     ...
|   |     +--rw geographic-location
|   |        +--rw description?   string
|   |        +--rw source*        string
|   |        +--rw destination*   string
```

- **Note**
  - The registration of a pair of (name, value) for a condition in CFI needs to be done to Security Controller by CFI YANG data model.
  - With this, Security Policy Translator can perform a policy translation.

Condition YANG Tree Comparison – Context (3/3)

- The YANG data model in CFI has context condition that can be one-to-one mapped context in NFI.
- CFI and NFI YANG data models provide time condition to define the active period of a rule.
- CFI and NFI YANG data models provide geographic location condition to filter traffic from/to a certain region. This can be mapped into the source and destination IP (IPv4 or IPv6) addresses based on the database provided.
- CFI provides the registration of IP (IPv4 or IPv6) addresses to the database with /i2nsf-cfi-policy/endpoint-groups/location-group/name

Action YANG Tree Comparison (1/2)

Consumer-Facing Interface (CFI):

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

Primary action: Ingress and Egress action
Secondary action: Log action

NSF-Facing Interface (NFI):

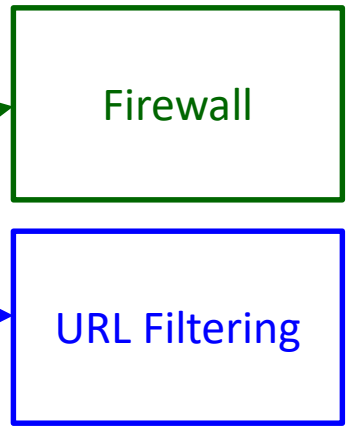
```
 |  +--rw action
 |     +--rw description?       string
 |     +--rw packet-action
 |     |  +--rw ingress-action?   identityref
 |     |  +--rw egress-action?    identityref
 |     |  +--rw log-action?       identityref
 |     +--rw flow-action
 |     |  +--rw ingress-action?   identityref
 |     |  +--rw egress-action?    identityref
 |     |  +--rw log-action?       identityref
 |     +--rw advanced-action
 |        +--rw content-security-control*    identityref
 |        +--rw attack-mitigation-control*   identityref
```

- The action in CFI YANG data model is separated into primary-action and secondary-action. Primary action is Ingress and Egress action (i.e., pass, drop, reject, rate-limit, mirror, invoke-signaling, tunnel-encapsulation, forwarding, and transformation)

- In NFI YANG data model, the advanced action is used to activate the Service Function Chaining (SFC) to apply multiple NSFs on network traffics. This does not exist in CFI as the CFI is used to provide a high-level action.
  - The action of a certain policy (e.g., a URL filtering with firewall) in CFI may require multiple NSFs.
  - The SFC of those NSFs is handled by NFI.

Action YANG Tree Comparison (2/2)

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



Conclusion

1. There is no translation problem from a CFI policy to an NFI policy.
   - Security Controller can handle some missing elements in a CFI policy.

2. The CFI YANG data model provides a high-level policy for easy configuration.
   - The YANG data model in CFI provides the registration of a pair (name, value) for easy configuration to be saved into a database.

3. The NFI YANG data model focuses on providing security policy configuration for NSFs as a low-level policy to be understood by them.