Service Function Chaining–Enabled I2NSF Architecture

(draft-hyun-i2nsf-triggered-steering-03)

Sangwon Hyun*, Jaehoon Paul Jeong, Jung-Soo Park, and Susan Hares

IETF 99, Prague, Czech
July 18, 2017
Introduction

- **Objective**
  - This document describes an architecture that integrates service function chaining (SFC) into the I2NSF framework to support packet forwarding between NSFs.

- **Motivation**
  - To support an advanced security action in the I2NSF framework that allows an NSF to call another type of NSF
  - To enable composite inspection of packets through various types of NSFs
  - To enable load balancing over multiple NSF instances combined with dynamic NSF instantiation
To trigger an advanced security action, NSF₁ appends the capability name required for the advanced security action in NSH.

NSH includes:
- Service Path Identifier (e.g., SPI=1)
- Service Index (e.g., SI=0)
- Capability name required for an advanced security action (e.g., DPI)

SPI 1: NSF₁
SPI 2: NSF₁ → NSF₂
SFC-based Packet Forwarding in I2NSF

- Identify the particular NSF for DPI (NSF₂ is a DPI.) specified in NSH and determine the new NSF path of the packet
- Re-classification to change the existing path into the new one (SPI=2, SI=1)

- The classifier may be co-resident with the NSFs.

**NSH includes**
- Service Path Identifier (e.g., SPI=1)
- Service Index (e.g., SI=0)
- NSF name required for an advanced security action (e.g., DPI)

**Re-classification request & response**

1. Classifier
2. SFF

- SPI 1: NSF₁
- SPI 2: NSF₁ → NSF₂
SFC-based Packet Forwarding in I2NSF

- SPI 1: NSF₁
- SPI 2: NSF₁ → NSF₂

Classifier

SFF

Packet forwarding

SPI = 2, SI = 1

Re-classification request & response

- Interpret the NSF path information
- Identify the next NSF on the path
- Forward the packet to the next NSF

<table>
<thead>
<tr>
<th>SPI</th>
<th>SI</th>
<th>NH</th>
<th>Transport protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
<td>10.1.1.2</td>
<td>GRE</td>
</tr>
</tbody>
</table>

10.1.1.2
Configuration for SFC

- The Security Controller configures the classifier with service function chain/path information.
- The Security Controller generates the forwarding information table of NSFs and configures the SFF with it.

- NSF path information
- Mappings between capability names and NSFs

SPI 1: NSF₁
SPI 2: NSF₁ → NSF₂

<table>
<thead>
<tr>
<th>SPI</th>
<th>SI</th>
<th>NH</th>
<th>Transport protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
<td>10.1.1.2</td>
<td>GRE</td>
</tr>
</tbody>
</table>
Discussion

• SFC is suitable for enforcing the default (pre-determined) NSF path.

• Re-classification is required to support an advanced security action that the next NSF is determined in the I2NSF framework.
  – Introducing some overhead particularly when the classifier exists separately from an NSF

• Identifying a particular NSF for the given capability name (e.g., DPI) is required to fit into the I2NSF framework.
  – Interface between the Security Controller and SFC component (e.g., classifier, SFF) is required. \( \rightarrow \) I2NSF-SFC Interface?
The following changes have been made from draft-hyun-i2nsf-nsf-triggered-steering-02.

- Sections 3 (Terminology), 4 (Architecture), and 5 (Use Case) have been revised to describe the integration of the I2NSF framework and SFC and the process of packet forwarding between NSFs.
Next Steps

• We will specify more details of what kind of information should be included in the NSH header to support packet forwarding between NSFIs and also the formats.

• Design of I2NSF-SFC Interface
  – Information model & data model