Service Function Chaining: Subscriber and Policy Identification Variable-Length NSH Context Headers


SFC WG Meeting
Bangkok, October 2018

Mohamed Boucadair, Dirk von Hugo, Behcet Sarikaya
Agenda

• The Problem
• The Solution
• Updates vs. draft-sarikaya-sfc-hostid-serviceheader
• What is Next?
On Policies

• Some service deployments require enforcing policies based on the internal IP address/prefix, a subscriber identifier, or a combination thereof.

• Also some services require differentiated policies (e.g., low latency, high resiliency, etc.)
  – SF invocation should be altered accordingly

• These policies may be enforced by one or multiple Service Functions

• These SFs may be located anywhere within an SFC-enabled domain

• The exact set of policies to be enforced are deployment-specific
The Problem

• A Service Function that needs to enforce policies **may not have access** to layer-2 subscriber Identifier (MAC@, Line ID, etc.)
  – Difficult to access to a Layer 2 information when the SF is located upstream

• Transport coordinates **are not reliable** for identifying a policy
  – Internal IP address is altered by on-path SFs (translators)
  – IP address/prefix assignment is dynamic

• **How to pass that information to upstream SFs for the sake of policy enforcement while avoiding relying on volatile information?**
The Solution

• **Pass** the “identification” data to be consumed by upstream SFs in a dedicated **NEW** context object
  
  – As part of the NSH header
  – Compliant with Section 4.9 of RFC 7665 (« Sharing metadata »)

• **Two context headers are specified:**
  
  – Subscriber Identifier
  – Policy Identifier

  can be included in the same NSH

• **Defined as Opaque data**
  
  – The structure of the data is deployment-specific
Changes vs draft-sarikaya-sfc-hostid-serviceheader

• **Address the comments** raised during the call for adoption
• **Simplify** the design by focusing on two key TLVs (subscriber and policy)
• **Abandon** defining a structure for the identifiers
• **Remove** all use cases and slice-related metadata
• Almost a complete re-write!
What’s Next?

• -01 is stable enough and addresses the comments received so far

• Consider a call for adoption