

Analysis on Forwarding Methods for Service Chaining

draft-homma-sfc-forwarding-methods-analysis-01

Shunsuke Homma (homma.shunsuke@lab.ntt.co.jp)

Kengo Naito (naito.kengo@lab.ntt.co.jp)

Diego R. Lopez (diego.r.lopez@telefonica.com)

Martin Stiernerling (ietf.stiernerling.org)

David Dolson (ddolson@sandvine.com) -- presenting

Purpose

- Investigate network design patterns and methods for service chaining
- Enable conversation and recommendations about solutions

Forwarding Methods

1. Flow-identifiable information (e.g., OpenFlow)
 - E.g., 5-tuple rules in every forwarder
 - Suitable for small or static networks
2. Stacked transport headers (e.g., SPRING)
 - Uses existing technologies, but packet size increases and fragmentation reduce network efficiency
3. Service Chain ID tags (e.g., NSH)
 - Good trade-off of packet size, forwarding table size for large networks
 - But requires new/upgraded equipment
 - This method is applicable to any size network, but is the only one we recommend for large-scale networks
(millions of customers, thousands of middle boxes)

Path Selection Patterns

1. Static: classify packets to network-wide end-to-end paths

- a) SF Shared: an SF can handle more than one path
- b) SF Dedicated: an SF handles only one path

Uses simple components but network-wide control can be complex.

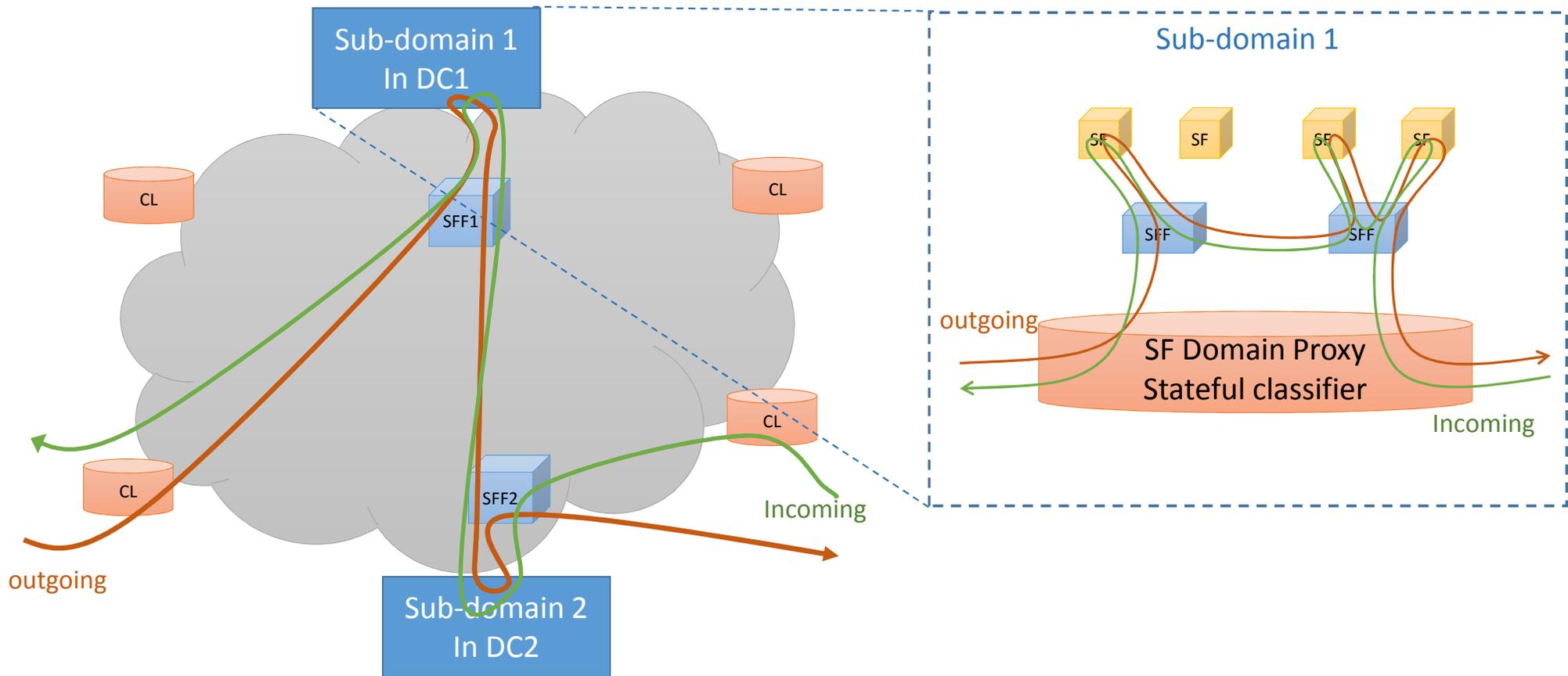
2. Dynamic: a logical chain is segmented into a series of segmented paths, with classification at each path segment

- Localized domains for fault management and control
- Stateful classification at multiple points

Hierarchical Service Paths

- An SF Sub-Domain can appear as a single SF to a high-level SF domain
- High-level SF domain:
 - Coarse classification
 - Relatively static paths
 - Widely distributed classifiers
- Low-level SF domain:
 - Stateful transport-session classification, DPI, dynamic network policy
 - Co-located classifiers to consistently handle bidirectional sessions
 - Co-located SFs to handle chatty control-plane, NFV elasticity

Hierarchical Service Paths



SF-Domain Proxy

- We named the module linking the domains “SF Domain Proxy”
 - We think we will rename it to “SF Domain Gateway”
- Looks like an SF to top level
- Looks like classifier and end-of-chain to low level

Benefits:

- Avoid costly stateful classification at distributed classifiers
 - Scales to very large networks
 - Supports specialized sub-domains with local control (e.g., per tenant)
- Inclusion of SF Domain Gateway in SFC Architecture?

Next Steps

- We believe the draft should progress, to assist operators
 - We would like to hear more examples
- “SF Domain Gateway” inclusion in SFC architecture?
- We are interested in creating a new draft for the hierarchical approach.