Network Working Group
Internet-Draft
Intended status: Informational
Expires: October 3, 2014

P. Quinn, Ed.
Cisco Systems, Inc.
T. Nadeau, Ed.
Brocade
April 1, 2014

Service Function Chaining Problem Statement
draft-ietf-sfc-problem-statement-03.txt

Abstract

This document provides an overview of the issues associated with the deployment of service functions (such as firewalls, load balancers) in large-scale environments. The term service function chaining is used to describe the definition and instantiation of an ordered set of instances of such service functions, and the subsequent "steering" of traffic flows through those service functions.

The set of enabled service function chains reflect operator service offerings and is designed in conjunction with application delivery and service and network policy.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on October 3, 2014.

Table of Contents

1. Introduction 3
1.1. Definition of Terms 3
2. Problem Space 5
2.1. Topological Dependencies 5
2.2. Configuration complexity 5
2.3. Constrained High Availability 6
2.4. Consistent Ordering of Service Functions 6
2.5. Application of Service Policy 6
2.6. Transport Dependence 7
2.7. Elastic Service Delivery 7
2.8. Traffic Selection Criteria 7
2.9. Limited End-to-End Service Visibility 7
2.10. Per-Service (re)Classification 7
2.11. Symmetric Traffic Flows 8
2.12. Multi-vendor Service Functions 8
3. Service Function Chaining 9
3.1. Service Overlay 9
3.2. Control Plane 9
3.3. Service Classification 9
3.4. Dataplane Metadata 10
4. Related IETF Work 11
5. Summary 12
6. Security Considerations 13
7. Contributors 14
8. Acknowledgments 16
9. Informative References 17
Authors' Addresses 18

1. Introduction

The delivery of end-to-end services often require various service functions including traditional network service functions (for example firewalls and server load balancers), as well as application-specific features. Service functions may be delivered within the context of an isolated user group, or shared amongst many users/user groups

Current service function deployment models are relatively static in that they are tightly coupled to network topology and physical resources. The result of that static nature of existing deployments greatly reduces, and in many cases, limits the ability of an operator to introduce new services and/or service functions. Furthermore there is a cascading effect: service changes affect other services.

This document outlines the problems encountered with existing service deployment models for Service Function Chaining (SFC) (often referred to simply as service chaining; in this document the terms will be used interchangeably), as well as the problems of service chain creation/ deletion, policy integration with service chains, and policy enforcement within the network infrastructure.

1.1. Definition of Terms

Classification: Locally instantiated policy that results in matching of traffic flows for identification of appropriate outbound forwarding actions.

Network Overlay: A logical network built, via virtual links or packet encapsulation, over an existing network (the underlay).

Network Service: An externally visible service offered by a network operator; a service may consist of a single service function or a composite built from several service functions executed in one or more pre-determined sequences and delivered by one or more service nodes.

Service Function: A function that is responsible for specific treatment of received packets. A Service Function can act at the network layer or other OSI layers. A Service Function can be a virtual instance or be embedded in a physical network element. One of multiple Service Functions can be embedded in the same network element. Multiple instances of the Service Function can be enabled in the same administrative domain.

A non-exhaustive list of Service Functions includes: firewalls, WAN and application acceleration, Deep Packet Inspection (DPI),

server load balancers, NAT44 [RFC3022], NAT64 [RFC6146], HOST_ID injection [RFC6967], HTTP Header Enrichment functions, TCP optimizer, etc.

The generic term "L4-L7 services" is often used to describe many service functions.

Service Function Chain (SFC): A service Function chain defines an ordered set of service functions that must be applied to packets and/or layer-2 frames selected as a result of classification. The implied order may not be a linear progression as nodes may copy to more than one branch. The term service chain is often used as shorthand for service function chain.

Service Function Path (SFP): The instantiation of a service function chain in the network. Packets follow a service function path from a classifier through the required instances of service functions in the network.

Service Node (SN): Physical or virtual element that hosts one or more service functions.

Service Overlay: An overlay network created for the purpose of forwarding data along a service function path.

Service Topology: The service overlay connectivity forms a service topology.

2. Problem Space

The following points describe aspects of existing service deployments that are problematic, and that the Service Function Chaining (SFC) working group aims to address.

2.1. Topological Dependencies

Network service deployments are often coupled to network topology, whether it be real or virtualized, or a hybrid of the two. Such dependency imposes constraints on the service delivery, potentially inhibiting the network operator from optimally utilizing service resources, and reduces the flexibility. This limits scale, capacity, and redundancy across network resources.

These topologies serve only to "insert" the service function (i.e., ensure that traffic traverses a service function); they are not required from a native packet delivery perspective. For example, firewalls often require an "in" and "out" layer-2 segment and adding a new firewall requires changing the topology (i.e., adding new layer-2 segments).

As more service functions are required - often with strict ordering - topology changes are needed before and after each service function resulting in complex network changes and device configuration. In such topologies, all traffic, whether a service function needs to be applied or not, often passes through the same strict order.

The topological coupling limits placement and selection of service functions: service functions are "fixed" in place by topology and therefore placement and service function selection taking into account network topology information is not viable. Furthermore, altering the services traversed, or their order, based on flow direction is not possible.

A common example is web servers using a server load balancer as the default gateway. When the web service responds to non-load balanced traffic (e.g., administrative or backup operations) all traffic from the server must traverse the load balancer forcing network administrators to create complex routing schemes or create additional interfaces to provide an alternate topology.

2.2. Configuration complexity

A direct consequence of topological dependencies is the complexity of the entire configuration, specifically in deploying service function chains. Simple actions such as changing the order of the service functions in a service function chain require changes to the

topology. Changes to the topology are avoided by the network operator once installed, configured and deployed in production environments fearing misconfiguration and downtime. All of this leads to very static service delivery deployments. Furthermore, the speed at which these topological changes can be made is not rapid or dynamic enough as it often requires manual intervention, or use of slow provisioning systems.

2.3. Constrained High Availability

An effect of topological dependency is constrained service function high availability. Worse, when modified, inadvertent non-high availability or downtime can result.

Since traffic reaches many service functions based on network topology, alternate, or redundant service functions must be placed in the same topology as the primary service.

2.4. Consistent Ordering of Service Functions

Service functions are typically independent; service function_1 (SF1)...service function_n (SFn) are unrelated and there is no notion at the service layer that SF1 occurs before SF2. However, to an administrator many service functions have a strict ordering that must be in place, yet the administrator has no consistent way to impose and verify the ordering of the service functions that are used to deliver a given service.

Service function chains today are most typically built through manual configuration processes. These are slow and error prone. With the advent of newer service deployment models the control and policy planes provide not only connectivity state, but will also be increasingly utilized for the creation of network services. Such a control/management planes could be centralized, or be distributed.

2.5. Application of Service Policy

Service functions rely on topology information such as VLANs or packet (re) classification to determine service policy selection, i.e. the service function specific action taken. Topology information is increasingly less viable due to scaling, tenancy and complexity reasons. The topological information is often stale, providing the operator with inaccurate placement that can result in suboptimal resource utilization. Furthermore topology-centric information often does not convey adequate information to the service functions, forcing functions to individually perform more granular classification.

2.6. Transport Dependence

Service functions can and will be deployed in networks with a range of transports, including under and overlays. The coupling of service functions to topology requires service functions to support many transport encapsulations or for a transport gateway function to be present.

2.7. Elastic Service Delivery

Given that the current state of the art for adding/removing service functions largely centers around VLANs and routing changes, rapid changes to the service deployment can be hard to realize due to the risk and complexity of such changes.

2.8. Traffic Selection Criteria

Traffic selection is coarse, that is, all traffic on a particular segment traverse service functions whether the traffic requires service enforcement or not. This lack of traffic selection is largely due to the topological nature of service deployment since the forwarding topology dictates how (and what) data traverses service function(s). In some deployments, more granular traffic selection is achieved using policy routing or access control filtering. This results in operationally complex configurations and is still relatively inflexible.

2.9. Limited End-to-End Service Visibility

Troubleshooting service related issues is a complex process that involve both network-specific and service-specific expertise. This is especially the case when service function chains span multiple DCs, or across administrative boundaries. Furthermore, the physical and virtual environments (network and service), can be highly divergent in terms of topology and that topological variance adds to these challenges.

2.10. Per-Service (re)Classification

Classification occurs at each service function independent from previously applied service functions. More importantly, the classification functionality often differs per service function and service functions may not leverage the results from other service functions.

2.11. Symmetric Traffic Flows

Service function chains may be unidirectional or bidirectional depending on the state requirements of the service functions. In a unidirectional chain traffic is passed through a set of service functions in one forwarding direction only. Bidirectional chains require traffic to be passed through a set of service functions in both forwarding directions. Many common service functions such as DPI and firewall often require bidirectional chaining in order to ensure flow state is consistent.

Existing service deployment models provide a static approach to realizing forward and reverse service function chain association most often requiring complex configuration of each network device throughout the SFC.

2.12. Multi-vendor Service Functions

Deploying service functions from multiple vendors often require per-vendor expertise: insertion models differ, there are limited common attributes and inter- vendor service functions do not share information.

3. Service Function Chaining

Service Function Chaining aims to address the aforementioned problems associated with service deployment. Concretely, the SFC working group will investigate solutions that address the following elements:

3.1. Service Overlay

Service function chaining utilizes a service specific overlay that creates the service topology. The service overlay provides service function connectivity and is built "on top" of the existing network topology and allows operators to use whatever overlay or underlay they prefer to create a path between service functions, and to locate service functions in the network as needed.

Within the service topology, service functions can be viewed as resources for consumption and an arbitrary topology constructed to connect those resources in a required order. Adding new service functions to the topology is easily accomplished, and no underlying network changes are required.

Lastly, the service overlay can provide service specific information needed for troubleshooting service-related issues.

3.2. Control Plane

Service aware control plane(s) provide information about the available service functions on a network. The information provided by the control plane includes service network location (for topology creation), service type (e.g. firewall, load balancer, etc.) and, optionally, administrative information about the service functions such as load, capacity and operating status. The service aware control plane allows for the formulation of service function chains and exchanges requisite information needed to instantiate the service function chains in the network.

Furthermore, the service aware control plane may interact with the topology aware control plane (if separate) to ensure optimal selection (and possibly placement) of service function within a service function path.

3.3. Service Classification

Classification is used to select which traffic enters a service overlay. The granularity of the classification varies based on device capabilities, customer requirements, and service offered. Initial classification determines the service function chain required to process the traffic. Subsequent classification can be used within

a given service function chain to alter the sequence of service functions applied. Symmetric classification ensures that forward and reverse chains are in place. Similarly, asymmetric -- relative to required service function -- chains can be achieved via service classification.

3.4. Dataplane Metadata

Data plane metadata provides the ability to exchange information between classification entities and service functions, between service functions, service functions and classification entities and service nodes, and as such does not provide forwarding information used to deliver packet along the service overlay.

Metadata can include the result of antecedent classification, information from external sources or forwarding related data. Service functions utilize metadata, as required, for localized policy decisions.

In addition to sharing of information, the use of metadata addresses several of the issues raised in section 2, most notably the de-coupling of policy from the topology, and the need for per-service classification (and re-classification).

A common approach to service metadata creates a common foundation for interoperability between service functions, regardless of vendor.

4. Related IETF Work

The following subsections discuss related IETF work and are provided for reference. This section is not exhaustive, rather it provides an overview of the various initiatives and how they relate to network service chaining.

1. [L3VPN]: The L3VPN working group is responsible for defining, specifying and extending BGP/MPLS IP VPNs solutions. Although BGP/MPLS IP VPNs can be used as transport for service chaining deployments, the SFC WG focuses on the service specific protocols, not the general case of VPNs. Furthermore, BGP/MPLS IP VPNs do not address the requirements for service chaining.

2. [LISP]: LISP provides locator and ID separation. LISP can be used as an L3 overlay to transport service chaining data but does not address the specific service chaining problems highlighted in this document.

3. [NVO3]: The NVO3 working group is chartered with creation of problem statement and requirements documents for multi-tenant network overlays. NVO3 WG does not address service chaining protocols.

4. [ALTO]: The Application Layer Traffic Optimization Working Group is chartered to provide topological information at a higher abstraction layer, which can be based upon network policy, and with application-relevant service functions located in it. The mechanism for ALTO obtaining the topology can vary and policy can apply to what is provided or abstracted. This work could be leveraged and extended to address the need for services discovery.

5. [I2RS]: The Interface to the Routing System Working Group is chartered to investigate the rapid programming of a device's routing system, as well as the service of a generalized, multi-layered network topology. This work could be leveraged and extended to address some of the needs for service chaining in the topology and device programming areas.

6. [ForCES]: The ForCES working group has created a framework, requirements, a solution protocol, a logical function block library, and other associated documents in support of Forwarding and Control Element Separation. The work done by ForCES may provide a basis for both the separation of SFC elements, as well as provide protocol and design guidance for those elements.

5. Summary

This document highlights problems associated with network service deployment today and identifies several key areas that will be addressed by the SFC working group. Furthermore, this document identifies four components that are the basis for service function chaining. These components will form the areas of focus for the working group.

6. Security Considerations

Security considerations are not addressed in this problem statement only document. Given the scope of service chaining, and the implications on data and control planes, security considerations are clearly important and will be addressed in the specific protocol and deployment documents created by the SFC WG group.

7. Contributors

The following people are active contributors to this document and have provided review, content and concepts (listed alphabetically by surname):

Puneet Agarwal
Broadcom
Email: pagarwal@broadcom.com

Mohamed Boucadair
France Telecom
Email: mohamed.boucadair@orange.com

Abhishek Chauhan
Citrix
Email: Abhishek.Chauhan@citrix.com

Uri Elzur
Intel
Email: uri.elzur@intel.com

Kevin Glavin
Riverbed
Email: Kevin.Glavin@riverbed.com

Ken Gray
Cisco Systems
Email: kegray@cisco.com

Jim Guichard
Cisco Systems
Email:jguichar@cisco.com

Christian Jacquenet
France Telecom
Email: christian.jacquenet@orange.com

Surendra Kumar
Cisco Systems
Email: smkumar@cisco.com

Nic Leymann
Deutsche Telekom
Email: n.leymann@telekom.de

Darrel Lewis
Cisco Systems

Email: darlewis@cisco.com

Rajeev Manur
Broadcom
Email:rmanur@broadcom.com

Brad McConnell
Rackspace
Email: bmcconne@rackspace.com

Carlos Pignataro
Cisco Systems
Email: cpignata@cisco.com

Michael Smith
Cisco Systems
Email: michsmit@cisco.com

Navindra Yadav
Cisco Systems
Email: nyadav@cisco.com

8. Acknowledgments

The authors would like to thank David Ward, Rex Fernando, David Mcdysan, Jamal Hadi Salim, Charles Perkins, Andre Beliveau, Joel Halpern and Jim French for their reviews and comments.

9. Informative References

[ALTO] "Application-Layer Traffic Optimization (alto)", <http://datatracker.ietf.org/wg/alto/>.

[ForCES] "Forwarding and Control Element Separation (forces)", <http://datatracker.ietf.org/wg/forces/>.

[I2RS] "Interface to the Routing System (i2rs)", <http://datatracker.ietf.org/wg/i2rs/>.

[L3VPN] "Layer 3 Virtual Private Networks (l3vpn)", <http://datatracker.ietf.org/wg/l3vpn/>.

[LISP] "Locator/ID Separation Protocol (lisp)", <http://datatracker.ietf.org/wg/lisp/>.

[NVO3] "Network Virtualization Overlays (nvo3)", <http://datatracker.ietf.org/wg/nvo3/>.

[RFC3022] Srisuresh, P. and K. Egevang, "Traditional IP Network Address Translator (Traditional NAT)", RFC 3022, January 2001.

[RFC6146] Bagnulo, M., Matthews, P., and I. van Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", RFC 6146, April 2011.

[RFC6967] Boucadair, M., Touch, J., Levis, P., and R. Penno, "Analysis of Potential Solutions for Revealing a Host Identifier (HOST_ID) in Shared Address Deployments", RFC 6967, June 2013.

Authors' Addresses

Paul Quinn (editor)
Cisco Systems, Inc.

Email: paulq@cisco.com

Thomas Nadeau (editor)
Brocade

Email: tnadeau@lucidvision.com