SFC
Internet Draft
Intended status: Standards Track
Expires: September 3, 2014

C. Huang
Carleton University
Jiafeng Zhu
Huawei
Peng He
Ciena
Mar 3, 2014

Service Forwarding Label
draft-huang-sfc-service-forwarding-label-00.txt

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), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

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

Copyright Notice

Abstract

New services such as network function virtualization (NFV), service chaining, and application-centric traffic steering bring new opportunities for network providers and service providers. This internet draft defines a new Layer 5 packet header format called Service Forwarding Label (SFL) and procedures which can be used as a universal label to differentiate various services and forward packets based on different service requirements.

Table of Contents

1. Introduction

New services such as network virtualization, service chaining, and application-centric traffic steering bring new opportunities to network providers and service providers.

One of the primary new services that have been envisioned is the network virtualization service, which allows physical network provider to sell different virtual networks to different service network providers. Each service network provider can use its virtual network just like the way it uses its own private network while sharing underlying physical network resources with other service network providers. The physical network provider, on the other hand, can enjoy new revenue growth through selling virtual networks with different granularities.

Traditionally a service chain consists of a set of dedicated network service boxes such as firewall, load balancers, and application delivery controllers that are concatenated together to support a

specific application. With a new service request, new devices must be installed and interconnected in certain order. This can be a very complex, time-consuming, and error-prone process, requiring careful planning of topology changes and network outages and incurring high OPEX. This situation is exacerbated when a tenant requires different service sequences for different traffic flows or when multiple tenants share the same datacenter network.

Network Function Virtualization (NFV) is a concept built upon network virtualization. It involves the implementation of network functions in software that can run on a range of industry standard high volume servers, switches, and storage. Through NFV, service providers can dynamically create a virtual environment for a specific service chain and eliminate the dedicated hardware and complex labor work for supporting a new service chain request.

Both network virtualization and NFV require traffic steering. However there are many other applications that can be better served with application-centric traffic steering. Application service providers have tried various ways to differentiate their customers so that they can maximize their revenues and minimize their costs. For example, cookies have been used to track HTTP users. Unfortunately they are designed for specific applications. Because cookies only appear in HTTP headers, they will not be carried by all packets except the first one. Therefore they cannot be used by switches to steer traffic. On the other hand, DiffServ and VLAN sit below Layer 4, they are hard to be maintained end-to-end. Therefore application-centric traffic steering, although widely desired, is still hard to achieve.

In order to support various services, a universal ID that can be used to identify a service instance (or a service chain instance) is required. This ID needs to sit above Layer 4 so that it can stay intact while a packet traverses legacy IP networks and middle-boxes. This naturally points to an ID at Layer 5.

In OSI model, Layer 5 is called the session layer which is designed to establish, manage, and terminate connections between local and remote applications. A good example is a video conference session where multiple parties join and leave dynamically. This bears similarity to a service instance such as a virtual network which carries a large number of dynamic traffic flows. This similarity is the motivation to define an ID at Layer 5 for the identification of a service instance (or service chain instance). We call this ID Service Forwarding Label (SFL).

In this document, the format of SFL is defined and procedures for assigning and removing SFLs are described.

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119.

In this document, these words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying RFC-2119 significance.

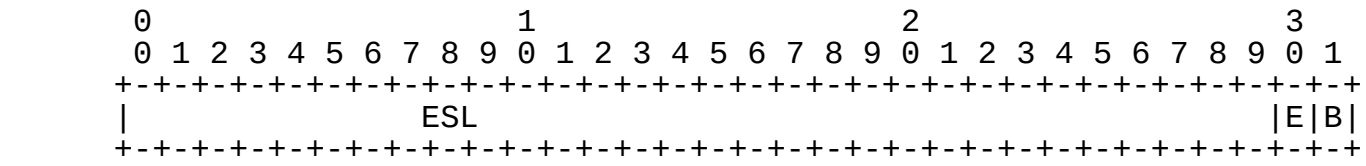
3. Formal Syntax

An SFL SHOULD be created and maintained by a service provider and used by its clients. Network switches and steering boxes SHOULD use SFL in part or full to identify and steer traffic belonging to different service instances. Service instances SHOULD use SFL in part or full to identify different service requirements from clients. SFLs can be stacked for applications such as recursive services where each level of the stack is administered by the owner of the service level in a recursive business relationship. This allows easy scale to multiple levels of services with multiple ownerships nested in the SFL stack.

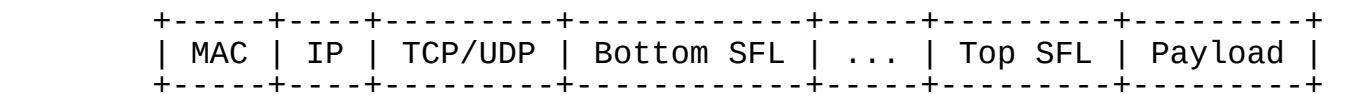
An SFL MUST be unique within the space of the service provider who administers the SFL. Multiple service providers at the same level will be differentiated by their SFLs assigned by their lower level service provider. The combination of the SFLs across different levels in a label stack uniquely identifies a service in a physical substrate domain.

As shown in Fig. 1 (a), each SFL is represented by 4 octets. Starting from bit 0 of the 4 octets, the first 30 bits hold the label, bit 30 is reserved for experimental use (E), bit 31 is the top-of-stack bit (T). The T bit is set to one for the top entry in a label stack, and zero for all other label entries in the stack. As the header at Layer 5, SFLs can run either over UDP or TCP making it applicable to all kinds of traffic belonging to the same service instance. A unique port number for both UDP and TCP SHALL be assigned to identify the existence of SFLs. It is RECOMMENDED that the top entry in an SFL stack SHOULD be used to identify different applications. A sample format for a packet with SFLs is shown in Fig. 1 (b).

Each SFL is associated with a lifetime. When its lifetime expires, the SFL SHALL be terminated or be renewed. This dynamic mechanism allows a service provider to maintain a smaller pool of SFLs.



(a)



(b)

Figure 1 (a) SFL format. (b)Packet with SFL.

4. Procedures

There are various scenarios that may happen during the lifetime of an SFL. The procedures for establishing and terminating SFL depend on the actual scenario encountered. The procedures are described step by step in the following part. For the description of simplicity, it is assumed that switches be OpenFlow enabled. SFL can be applied to other types of switches or steering boxes.

- o A client sends a service request to a service provider with its user ID and requested service type using HTTP request message. Metadata can be sent through HTTP Post message.

- o The service provider decides whether it can accept the request by applying optimization process which determines how to route traffic and allocate resources for the requested service.

- o If the request is admissible, the service provider will create a new SFL which is unique to the service provider and send the SFL and associated lifetime to switches within its substrate domain or middle-boxes that need to steer or process traffic based on the SFL through an OpenFlow OFPT_FLOW_MOD message.

- o Upon receiving the message, the OpenFlow switches or middle-boxes will set the SFL and its lifetime into their flow tables as part of a rule set.

- The service provider will send HTTP response message with the SFL and associated lifetime to the client confirming the acceptance of the request.

- The client will add the label as Layer 5 header to its packets destined for the requested service and send them out.

- When the packets reach the switches or middle-boxes within the service provider network, the service provider will match the Layer 5 header (and other headers in other layers if necessary) to its rule set and decide how to forward or process the packets based on their service requirement.

- The switches or middle-boxes will then process those packets and steer them to the next switch or middle-box if necessary.

- When the lifetime of the SFL expires, the client can choose either to renew the service or leave. If it decides to renew, it will send a HTTP request message with the SFL to the OFC, the above procedures will be repeated except that the original SFL will be used instead of generating a new SFL.

5. Use Cases

There are numerous use cases that SFL can be applied to. Some common use cases are briefly described in this section.

5.1. Network Virtualization

The first use case is network virtualization service. Here a physical network provider will serve as the service provider and virtual network providers will serve as clients. Virtual network providers request virtual networks from the physical network provider. Each virtual network provider will have full control over its virtual network. One issue is that the address space used by virtual network providers can be overlapped. For example, Client 1 owns Virtual Network 1 and Client 2 owns Virtual Network 2. Both Virtual Network 1 and Virtual Network 2 share a physical network owned by a infrastructure network provider. When a packet reaches a switch in the physical network domain, the switch needs to decide which virtual network the packet belongs to.

Through the procedures discussed in the last section, each client network will receive an SFL assigned by the infrastructure network provider as an identifier of its virtual network. The client network will inform its users of adding the SFL for all packets that need to use the virtual network it owns. When packets reach the switches in

the physical network domain, they can be differentiated using SFL even though their IP address spaces may be overlapped. Without SFL, multiple header fields may need to be matched in order to identify packets belonging to a virtual network, which will likely cause flow table fragmented and bloated.

When recursive network virtualization is deployed, each virtual network provider will serve as client as well as service provider at the same time. As a client, it receives an SFL from the service provider one level below it. As the service provider, it administers the SFLs that identify the virtual networks it sells. A physical switch can use multiple levels of the label stack to steer packets to the correct virtual networks they belong to.

5.2. Service Chaining

The second use case demonstrates how service chaining, as an example of NFV, can be supported.

Consider a scenario where an enterprise leases a virtual network from an infrastructure provider and provides two types of service chains. The first service chain, designed for its employees, will force traffic flows to go through NAT (network address translation), DPI (deep packet inspection), firewall, LB (load balancer), and various servers. The second one, designed for guest visitors, will only go through NAT and web servers. Each service chain is assigned an SFL by the enterprise while the virtual network of the enterprise will be assigned an SFL by the infrastructure provider. Traffic flows for different service chain instances can be uniquely identified and steered by the combinations of the two SFLs (one by the infrastructure provider and one by the enterprise). Within a service chain, each virtual node represents a specific function such as firewall that can be dynamically mapped to a physical node in the lower level. By the virtualization of a service chain, dynamic sharing of physical resources can be achieved. This enables great flexibility and leads to significant cost reduction in OPEX.

5.3. Application-Centric Traffic Steering

Service providers are increasingly interested in providing different treatments to different types of customers, e.g. subscribers vs. casual users. Based on the SFLs they are carrying, user traffic flows can be steered to different environments with different networking and computing resources provisioned. Under this context, SFL provides a simple and effective handle that connects applications to physical layer devices directly and enables application-centric traffic steering. There are many existing

Quality of Service (QoS) schemes such as VLAN and DiffServ. But they are Layer 2 or 3 mechanisms which are hard to scale to end-to-end applications. As mentioned earlier, it is difficult to maintain any code points in headers up to Layer 4 for end-to-end services due to middle boxes and different domains a packet may traverse. By sitting at Layer 5, SFL can travel through networks and middle boxes easily and therefore provide a very strong support for various end-to-end applications.

There are many application scenarios that can demonstrate the usage of SFL. For example, a service provider may want some of its user traffic be protected from server or link failures while other traffic not. When a server or link failure happens, the traffic that needs protection is steered to a protection path. The SFL provides an excellent option to achieve this function. Specifically, assign one SFL to identify traffic requiring protection and another SFL for traffic not requiring protection. When packets arrive at a switch, if the SFL matching indicates a packet without protection requirement, other header fields will be matched as regular case; otherwise, the packets will be forwarded to a group table for protection matching.

6. Migration

When networks that support SFL form some islands, due to the fact that SFLs sit in Layer 5, packets carrying SFLs will travel through the legacy network just like regular packets while being directed to their requested service instances in SFL enabled networks. This allows SFL enabled networks to coexist with legacy Internet.

7. Security Considerations

For security concern, HTTPS SHOULD be used for the creation and termination of SFLs. It is not recommended to use SSL in transport layer because this may cause difficulty for matching SFLs at a switch. However if SFLs are purely created for service chains, SSL MAY still be used as transport layer. In either case, a certificate MAY be created and attached to the SFL stack to ensure the integrity of the SFLs in the stack.

8. IANA Considerations

It is recommended that IANA assign a port in UDP and another port number in TCP to identify the existing of SFLs in Layer 5. The top level SFL of a SFL stack can use all existing port number assignments to identify various applications.

Authors' Addresses

Changcheng Huang
Department of Systems and Computer Engineering
Carleton University
1125 Colonel By Drive
Ottawa, ON K1S 5B6
Canada
Email: huang@sce.carleton.ca

Jiafeng Zhu
Huawei Technologies Inc
Santa Clara, CA
US
Email: Jiafeng.zhu@huawei.com

Peng He
Ciena Corp
Email: phe@ciena.com