SFC WG
Internet-Draft
Intended status: Informational
Expires: December 6, 2019

T. Ao
Individual
G. Mirsky
ZTE Corp.

June 4, 2019

Analysis of the SFC scalability
draft-ao-sfc-scalability-analysis-05

Abstract

SFC is an ordered set of service function, should be scalable to meet broad range of requirements. The scalability of SFC can be interpreted as ability of the SFC to accommodate one or more SFs joining the SFC , or leaving the SFC without significant impact to SFC performance.

This document presents four aspects on SFC scalability, and provide analysis of the data plane and the control plane to implement the scalable SFC.

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 https://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 December 6, 2019.

Table of Contents

1. Introduction

Service Function Chain (SFC) is the chain with a series of ordered Service Functions(SF). The SFC maybe changed because of load balance , failure, or other management requirement. We call it SFC scalability. The SFC being scalable means that the Service Functions can be added or removed from the path of this SFC without impact on other SFCs and minimal impact in the SFC being modified. With this capability, SFC is more flexible and elastic to adapt all kinds of requirements.

In this document, we will present four use cases on SFC scale-out and scale-in, and analysis some requirements to support SFC scalability.

2. Terminology

SFC(Service Function Chain): An ordered set of some abstract SFs.

SFC Scale-out: One or more SFs are added into the path of the SFC for the sake of load balance, protection or other new services requirement.

SFC Scale-in: One or more SFs are removed from the path of the SFC for the sake of the SFs are by-passed or the SFs are failed.

3. Four Use cases for scale-out/scale-in

Following describes four use cases to illustrate the scalability of the SFC.

3.1. Join

This is SFC horizontal scale-out use case. One or more new SFs must be added to a certain SFC for the traffic that has been classified to require application of new SF(s). This case is the reverse scenario to the by-pass. In this case one or more SFs that were by-passed need to be re-inserted into the SFC. And the SFC itself can be characterized as being scaled out.

There are two sub-cases of an SF joining the SFC. One when both the SF and corresponding SFF are new to the SFC. The second is when the SF attaches to an existing SFF. In the first scenario, control plane needs to notify the upstream SFF to modify its next hop to point to the new SFF and configure the new SFF's forwarding information. In the second scenario control plane needs to configure the existing SFF's forwarding information. In this scenario, SFF forwards the packets not only according to the SFPID but also according to the metadata in the SFC header.

3.2. Redundancy

3.2.1. SF Redundancy

This is an example of SFC vertical scale-out use case. One or more SFs are added into the SFC to meet the redundancy or load balance requirements for some certain SFs. This case is different from the Join case (section 3.1) in which the SF in this case is the same with one of the SF that is on the path of the SFC. The new SF have the same function with the existing SF, so that the new SF is added into the SFC to protect the existing corresponding SF and to load balance the existing corresponding SF. Figure 1 is the illustration about SF redundancy. In this figure, SF2' is the redundency of SF2, so that when SF2 is down, SF2' can keep working.

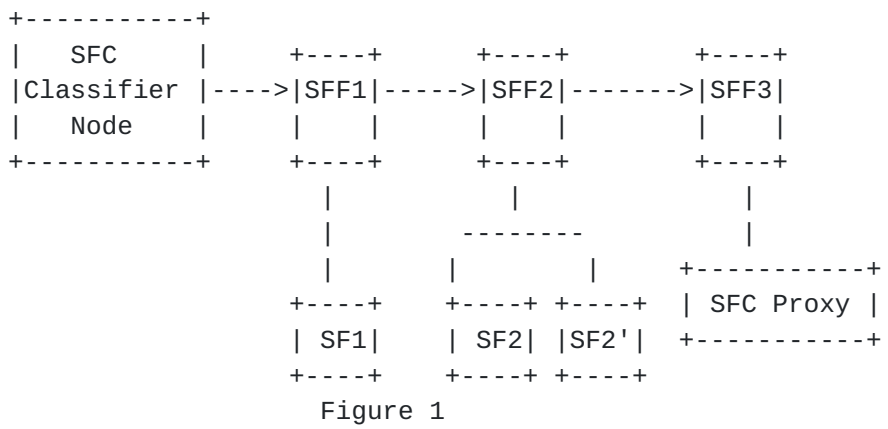

Figure 1

In this case, control plane need to notify the upstream SFF that a new SF joins the SFC as a redundancy SF for protection or load balance, and its next hop should be a protection group or ECMP group. For the purpose of load balance to ensure proper forwarding, the Flow Id field MUST be presented in the NSH as expression of entropy so that SFF can select an SF from the group according to the Flow Id. In the above figure, SFF2 knows that it is connecting a group of SFs and when it foward the packet, it would use Flow id in NSH.

3.2.2. SFC Redundancy

This is also an example of SFC vertical scal-out use case, namely Reduncancy. In this case, SFC is scaled out to two SFP paths. One SFP is redundant to another SFP, and the two SFPs are for protection or load balance. They belongs to a SFC, but have different SFP. The two SFPs are forming a group. Figure 2 is the illustration about the SFC redundancy. In this figure, we can see that SF1', SF2', SFC proxy' are the backup of the SF1, SF2, SFC Proxy seperately. The two SFPs are a group for the Classifier. All these nodes can be joint at some nodes and can be disjoint as well. In the figure 2, all the nodes are disjoint.

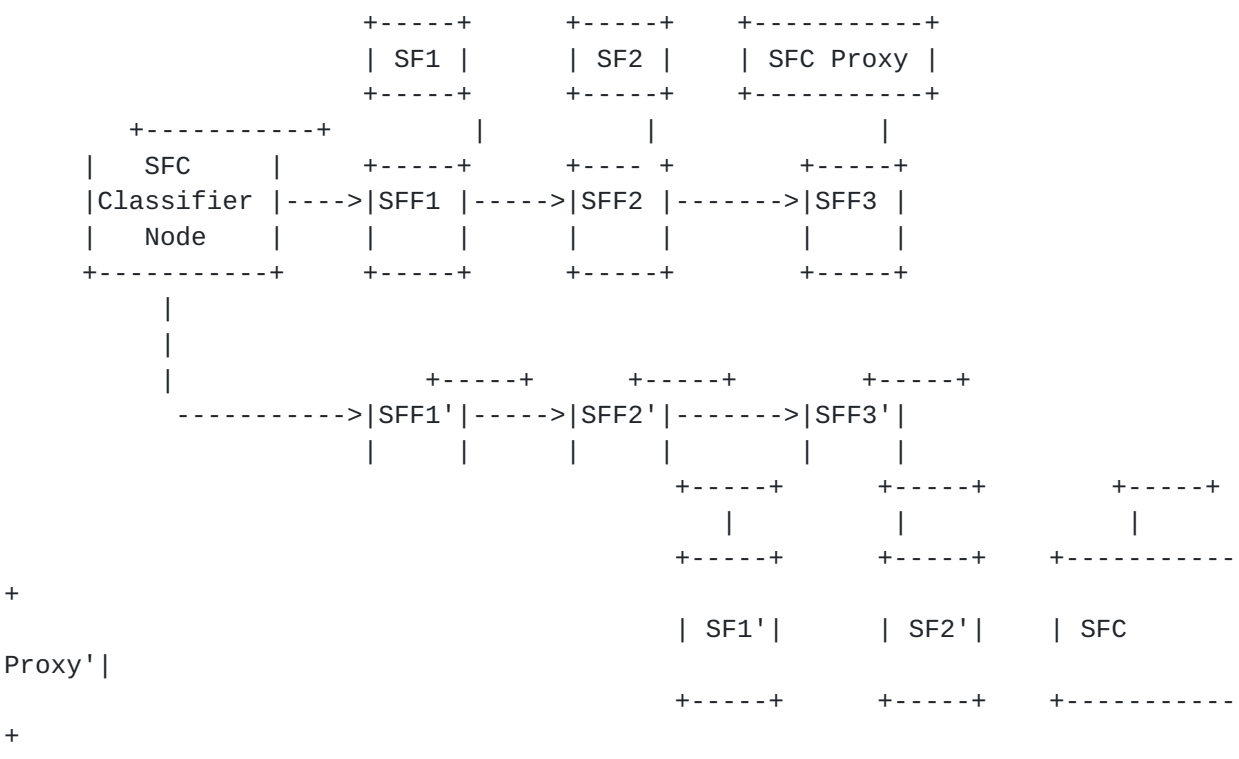

Figure 2

In this case, control plane need to notify the Classifier that the SFC is a group which contains two SFPs. The group can be used as protection or load balance. For the purpose of load balance, to ensure proper forwarding, the Flow Id field MUST be presented in the NSH as expression of entropy so that the forwarder in the classifier can select an SFP from the group according to the Flow Id. For the case of joint, the joint node also need to have capability to forward the traffic accroding to the Flow ID.

3.3. By-pass

This is an example of horizontal scale-in case. In this scenario some SFs are not removed from the SFC but just by-passed by the traffic so that the packets will not be processed by these SFs. Use cases for this scenario are described in [draft-ietf-sfc-long-lived-flow-use-cases] and [draft-ietf-sfc-offloads] . In these two drafts, the SF is offloaded because it is not necessary to steer the traffic to the SFs to improve the forwarding performance.

The corresponding solution is also provided in the above drafts.

3.4. Failure or Remove

This is a vertical SFC scale-in case. This happens only when the SFC

is being protected or load balanced. When SF of one SFC has failed or needs to be removed because it is no longer needed to do the pretection, the ability of the SFC to scale-in is excercised.

In this case, the upstream SFF MUST be notified that its next hop has been changed to the next SF of the SF.

From the cases described we can conclude that no matter if is SFC scale-out case or scale-in cases, there are some requirements to SFC control protocol. And for some cases, there are requirements to data plane as well.

4. Data Plane Requirements

For the cases of load balancing or protection switchover of SFC scalability, it is highly beneficial to have an entropy field in the SFC header NSH. The entropy may be presented in the dedicated field named as Flow ID which be part of SFC encapsulation.

This means that SFF not only forwards the traffic based on different SFPID, but also MAY use Flow ID to select particular SF out of set of SFs of the same type.

According to the NSH draft in draft--ietf-sfc-nsh-27, we propose to extend NSH to include the entropy field. Two options can be considered. One is to use existing field, for example, some reserved bits. Suggested extended field in NSH Service Path Header is showed in Figure 3.

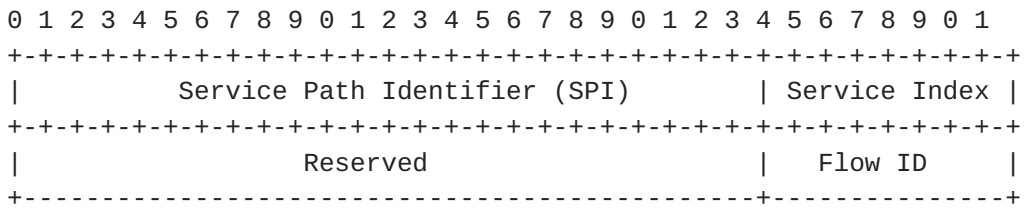

Figure 3

Another is to extend a new metadata to meet the requirement. Which has been described in the section 8 of the draft-quinn-sfc-nsh-tlv-04 .

5. Control Plane Requirements

5.1. Centralized CP

SFC Controller is required to:

a) Send a message to SFF that the joined SF connected to set the correct SFPID and its next hop.

b) Send register message to upstream SFF or classifier with some information. The information not only includes next hop locator, but

also includes an indicator if the next hop is a new joined SF or a group that a new SF that added into. If the indicator is a new joined SF, it means the new SF will join the SFC. If the indicator is a group, it means a new SF or a new SFP will be added into this group for load balance or protection.

c) Send de-register message to upstream SFF or classifier with some information. The information not only includes next hop locator, but also includes an indicator that if the next hop is by-passed, or the next hop is removed from a group. If the indicator is the by-passed SF, it means the current SF is by-passed or is leaving from the SFC. If the indicator is a group SF, it means the current SF or SFP will be removed from a protection group that is for load balance or protection.

5.2. Distributed CP

Distributed SFC CP can be used in Plug-and-Play scenario. Distributed SFC CP required:

a) The SF that needs to join into the SFC or be by-passed by the SFC should explicitly notify the SFF it is associated with.

b) Once get the connection notification from the SF, the associated SFF should send a register message to the upstream SFF with some information. Such information not only includes next hop locator, but also includes an indicator that if the next hop is a new joined SF or the next hop is a new SF that added into a group. If the indicator is a new joined SF, it means a new SF will join the SFC. If the indicator is a group, it means a new SF will be added into a group for load balance or protection.

c) The SFF send de-register message to upstream SFF with some information. Such information not only includes next hop locator, but also includes an indicator that if the next hop is the next SF because the current SF is by-passed, or the next hop is the SF that is removed from a group. If the indicator is the by-passed SF, it means the current SF is by-passed or is leaving from the SFC. If the indicator is group SF, it means the current SF will be removed into a protection group that is for load balance or protection.

6. Security Considerations

For the scalability of the SFC, security is very important to be considered. Before allow the SF to join to the SFC, it is required to make sure the SF's security first.

7. IANA Considerations

TBD

8. Information References

[I-D.ietf-sfc-architecture]
Halpern, J. and C. Pignataro, "Service Function Chaining (SFC) Architecture", draft-ietf-sfc-architecture-11 (work in progress), July 2015.

[I-D.ietf-sfc-long-lived-flow-use-cases]
Krishnan, R., Ghanwani, A., Halpern, J., Kini, S., and D. Lopez, "SFC Long-lived Flow Use Cases", draft-ietf-sfc-long-lived-flow-use-cases-03 (work in progress), February 2015.

[I-D.ietf-sfc-nsh]
Quinn, P., Elzur, U., and C. Pignataro, "Network Service Header (NSH)", draft-ietf-sfc-nsh-28 (work in progress), November 2017.

[I-D.ietf-sfc-offloads]
Kumar, S., Guichard, J., Quinn, P., Halpern, J., and S. Majee, "Service Function Simple Offloads", draft-ietf-sfc-offloads-00 (work in progress), April 2017.

[RFC7498] Quinn, P., Ed. and T. Nadeau, Ed., "Problem Statement for Service Function Chaining", RFC 7498, DOI 10.17487/RFC7498, April 2015, <https://www.rfc-editor.org/info/rfc7498>.

Authors' Addresses

Ting Ao
Individual
No.889, BiBo Road
Shanghai 201203
China

Phone: +86 17721209283
Email: 18555817@qq.com

Greg Mirsky
ZTE Corp.
1900 McCarthy Blvd. #205
Milpitas, CA 95035
USA

Email: gregimirsky@gmail.com