Network working group
Internet Draft
Intended status: Informational
Expires: January 2015

L. Dunbar
Huawei
Ron Parker
Affirmed Networks
I. Smith; S. Majee
F5 Networks
N. So
Vinci Systems
Donald Eastlake
Huawei

July 4, 2014

Architecture for Chaining Legacy Layer 4-7 Service Functions
draft-dunbar-sfc-legacy-l4-l7-chain-architecture-05.txt

Status of this Memo

Abstract

This draft describes the architecture for chaining existing Layer 4-7 service functions that are not aware of newly defined SFC header. The intent is to identify optimal architecture for flexibly chaining existing Layer 4-7 functions to meet various service needs.

Table of Contents

1. Introduction

This draft describes the architecture for chaining existing Layer 4-7 service functions that are not aware of newly defined SFC header. The intent is to identify optimal architecture for flexibly chaining existing Layer 4-7 functions to meet various service needs.

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 [RFC2119].

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.

Chain Classifier: A component that performs traffic classification and potentially encodes a unique identifier or the SF MAP Index introduced by [SFC-Framework] to the packets. The unique identifier in the packets can be used by other nodes to associate the packets to a specific service chain and/or steer the packets to the designated service functions.

DPI: Deep Packet Inspection

FW: Firewall

Legacy Layer 4-7 Service Function: Same as the Service Functions defined in [SFC-Problem] except that they may not be aware of the new service function chain header encapsulations. Many of existing Layer 4-7 service functions fall into this category. Exemplary functional modules include Firewall, Deep Packet Inspection (DPI), Encryption, Packet De-duplication, Compression, TCP Acceleration, NAT, and etc

Service Function Instance: One instantiation of a service function.

One service function could have multiple identical instances. For a service function with different functional instantiations, e.g. one instantiation applies policy-set-A (NAT44-A) and other applies policy-set-B (NAT44-B), they are considered as two different service functions."

Some Service Function Instances are visible to Service Chain Path. Sometimes a collection of service function instances can appear as one single entity to the Service Chain Path, leaving the instance selection to local nodes.

3. Legacy Layer 4-7 Service Functions and Chaining

Legacy Layer 4-7 service functions are the existing service functions that may not be aware of any new service encapsulation layers being proposed in SFC WG.

3.1. Layer 4-7 Service Functions

A Layer 4-7 service function performs certain action to the packets traversed through. By Layer 4-7, it means that those functions don't participate in network layer routing protocols. The implementation of such service function can be either Proxy based or Packet Based, or a hybrid of both when more than one function is performed to the same packet flow. Multiple service functions can be instantiated on a single service node as defined by [SFC-ARCH], or embedded in a L2/L3 network node.

- Proxy based service functions: these service functions terminate original packets, may reassemble multiple packets, reopen a new connection, or formulate new packets based on the received packets.

- Packet based service functions: these service functions maintain original packets, i.e. they don't make changes to packets traversed through except possibly making changes to metadata attached to the packet or the packet's outer header fields.

Some Layer 4-7 service functions might have intelligence to choose the subsequent service functions on a service chain and pass data packets directly to the selected service functions. However, most existing Layer 4-7 service functions don't have this capability.

3.2. Metadata to Layer 4-7 Service Functions

Strictly speaking, everything carrying the information that is not in the payload data is metadata. IETF has standardized many types of

metadata exchanged among L2/L3 nodes, e.g. QoS bits, MPLS labels, etc. Those metadata are out of the scope of SFC.

Metadata in the SFC sense must mean something more specific such as "the information added to the packet to be carried along with the packet for the consumption of the service function nodes along the chain".

This section classifies the metadata that are meaningful to SFC.

3.2.1. Metadata at different OSI Layers

- Application Layer metadata:

  Some Layer 4-7 service functions, especially the proxy based service functions, exchange metadata among themselves by changing the payload of the data packets, e.g. attaching a cookie to the payload or initiating a new TCP session.

  Those metadata, especially the metadata among L7 Service Functions, are considered as part of payload. Most likely they are proprietary to application layer. Therefore, they should be out of the scope of SFC.

- Layer 4-7 Service Function Layer Metadata

  Some service functions exchange information among themselves. Today, most of those metadata exchanges between legacy Layer 4-7 service functions are vendor specific.

- Network Layer metadata

  Some Layer 4-7 service functions exchange metadata with L2/L3 nodes to achieve desired network forwarding behavior.

3.2.2. Framework of carrying the metadata

- Message based metadata:

  Some service functions receive metadata from external entities (e.g. policy engines, controller, etc). In Mobile environment, some service functions receive metadata from PCRF via Diameter interfaces. Those metadata are normally flow based, e.g. applying

a specific QoS priority for data packets with specific Source/Destination Address(es), TCP port number, etc. Those metadata don't have to be attached to every data packet.

o Data Packet attached Metadata:

Some metadata has to be attached to packets to facilitate proper treatment by service functions.

o Hybrid Method:

Attaching extra metadata to every packet increases the likelihood of packet size exceeding MTU, which lead to packet fragmentation. Therefore, the metadata attached to packets have to be compact.

To reduce the metadata size attached to data packets, it is worth considering combining the "messaged based metadata" and the "Packet attached Metadata". I.e. attaching compact index to packets that can correlate to complete metadata passed down from separate messages from external systems.

4. Architecture for Chaining Legacy Layer 4-7 Service Functions

Chaining Layer 4-7 Service Functions not only needs the network that steers data flows to their designated service functions, but also needs an Service Chain Controller that can update the steering policies to the relevant forwarding nodes when changes occurs.

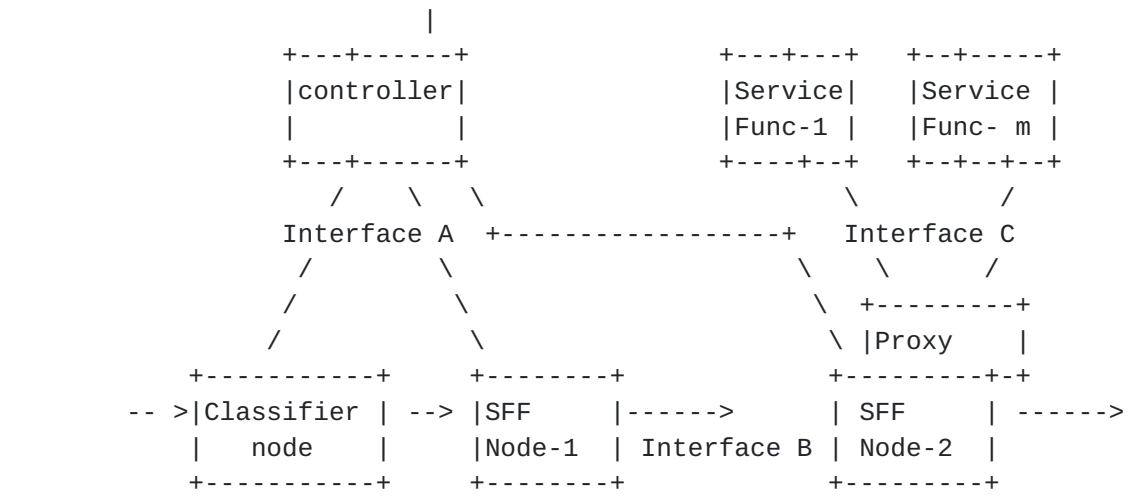

Figure 1 Interfaces needed for Chaining Service Functions

There are 3 types of interfaces to be addressed by the architecture:

- Interface A: this is the interface between the Service Chain controller and the relevant classifier/steering nodes to exchange the steering policies or/and other information for the service chains.

- Interface B: this is the network layer that transports the packets among SFF nodes. Proper tunnels might be needed among SFF nodes so that traffic can traverse the legacy network segments.

- Interface C: this is the interconnection between SFF function and Service Functions. Since some legacy SFs can't recognize the SFC header, a proxy entity is needed to convert the information extracted from SFC header to existing header or tags (e.g. VLANs) recognizable by the SFs for packets traversed on this interface.

4.1. Service Function Forwarder for Layer 4-7 Service Functions

For chaining together legacy Layer 4-7 service functions, the Service Function Forwarder (SFF) defined by [SFC-Arch] may need to terminate the service layer encapsulation on behalf of service functions/nodes that are not aware of the SFC header. There can be multiple SFF nodes in the Service Chain domains [SFC-Framework].

Even though Layer 4-7 Service functions can be instantiated anywhere, it is not uncommon to have multiple service functions instantiated on nodes in close vicinity to a Service Function Forwarder node. The following figure depicts the architecture for chaining those Layer 4-7 service nodes that are not aware of service layer encapsulation. Each SFF is responsible for steering the traffic to their designated local service functions and for forwarding the traffic to the next hop SFF after the local service functions treatment.

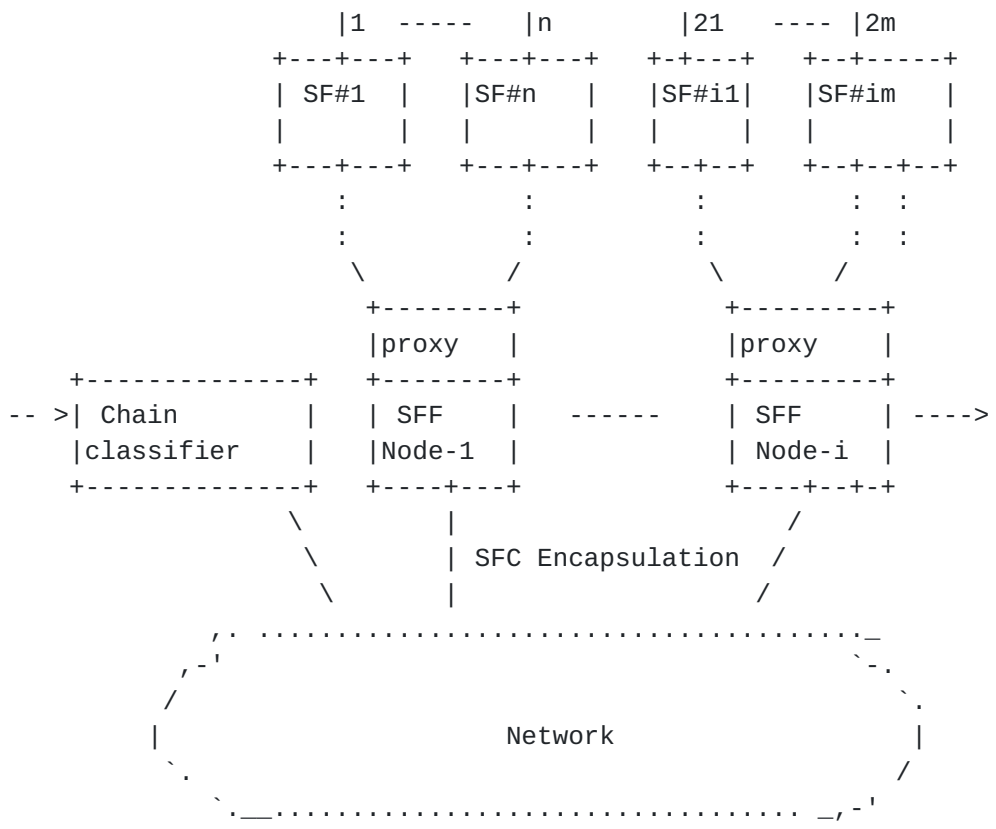

Figure 2 Chaining existing Layer 4-7 service nodes

The "Chain Classifier" node in the figure is to classify the incoming packets/frames into different service flows based on their service characteristics or policies from service chain orchestration or controller. Different service flows can be differentiated by some fields in the packets or can be encapsulated with the corresponding SFC header.

The steering policies for flows arriving at SFF Nodes can be carried by the SFC header in the data packets, separate out-of-band messages from Chain Classier or external controllers, or combination of both.

The SFF nodes can be standalone devices, or can be embedded within network forwarding nodes. Overlay tunnels are expected to connect the "SFF nodes" together.

4.2. Layer 4-7 nodes connection to SFF Nodes

Since the legacy SFs can't terminate the newly defined SFC header, there has to be a proxy entity either attached to or embedded in a SFF node. Here are the major responsibilities of the proxy entity:

- SFF-> SF direction:

  The proxy entity is needed to decapsulate the SFC header from the packets if the SFC header is not recognizable by the SF, extract the service chain identifier from the SFC header, map the service chain identifier to a locally significant tag or header that is recognizable by the legacy SF, and encapsulate the tag or the header to the data packets before sending the packets to the SF.

  By locally significant, it means that the tag or the header is only local to the link/path between the SFF Proxy entity and the SF, and is capable of differentiating packets from different service chains that traverse the link/path.

  Examples of locally significant tags include VLANs, GRE key, etc. Examples of locally significant header include encapsulating additional IP, MAC, or GRE header, etc.

  If there are metadata carried by the SFC header that are needed by the SF, the proxy entity is responsible for extracting the metadata from the SFC header and passing them to the Service Functions via a method that is supported by the Service Function.

- SF -> SFF direction:
  The proxy entity is responsible for constructing the SFC header expected by next SFF nodes from the locally significant tag/header when packets come back from the SF, encapsulating the SFC header back to the data packets before passing to the next SFF nodes.

Layer 4-7 Service nodes can be connected to SFF nodes in various ways. The topology could be bump in a wire or one armed topology.

- A service function can be embedded in a SFF node (i.e. embedded in a router or a switch). In this case, the combined entity forms the SF node described in the [SFC-ARCH].

- A service node can be one hop away from a SFF node

The one hop between the SFF node and the service node can be a physical link (e.g. Ethernet link). Under this scenario, there would be a Link Header, i.e. an outer MAC header, added to the data packets that meet the steering criteria.

The one hop link can be a transparent link, i.e. no link address is added to the data packets on the link between the SFF node and Service node. I.e. the service nodes can apply treatment to data frames arrived at the ingress port regardless of the Link Destination address.

- A service node can be multiple hops away, such as when a service function is deployed in an on-net or private *aaS offering. Under this scenario, a tunnel is needed between the service node and the SFF node.

4.3. Traffic Steering at SFF Nodes

The forwarding (or steering) policies for data packets received by the SFF Nodes can be carried by the SFC header in the data packets or combined with separate out-of-band messages from external controller(s) or the Chain Classifier. When using the out-of-band messages to carry the steering policies to SFF nodes, the steering policies have to be correlated with some fields in the data packets. Those fields of the data packets play the role of differentiating packets belong to different service chains.

It worth noting that when one SFF node have multiple Service Functions (SF) attached, there could be two different Chains going through one common SF#1, but the Chain #1 needs to go to SF#4 after SF#1, and the Chain #2 needs to go to another SFF node after the SF#1. The SFF node has to re-classify traffic coming back from a port connected to a SF if the Chain identifier is not carried by the data packets.

The policies to steer traffic to their corresponding service functions or service function instances can change. Those steering policies can be dynamically updated by SFC Header or the out-of-band messages.

There are many types of policies for SFF to steer data packets to their designated service functions, for example:

- Fixed header based forwarding: traffic steering based on header fields that have fixed position in the data packets:

- Forwarding based on Layer 2-3 header fields, such as MAC or IP Destination Address, Source Addresses, MPLS label, VLAN ID, or combination of multiple Layer 2-3 header fields.

- Forwarding based on Layer 4 header (TCP or UDP).

- QoS header based forwarding.

- Layer 7 based forwarding: traffic steering (or forwarding) based on the payload (L7) of data packets.

  Multiple data packets may carry some meaningful data, like one HTTP message. Under this scenario, multiple data packets have to be examined before meaningful data can be extracted for making Layer 7 based forwarding decision.

- Metadata based steering: traffic steering (or forwarding) based on the identity of the initiating user, the UE model or type, the site name or FQDN, or network conditions (congestion, utilization, etc.).

  However those metadata might not necessarily be carried by each data packet due to extended bits required that can cause high probability of packet fragmentation. Those metadata can be dynamically passed down to steering nodes in some forms of steering policies from network controller(s).

5. Control Plane for Layer 4-7 Service Function Chain

5.1. Multiple Instances of a Service Function

One service function could have multiple identical instances, potentially attached to different SFF nodes. It is also possible to have multiple identical service function instances attached to one Service Function Forwarder (SFF) node, especially in an environment where service function instances are running on virtual machines with each having limited capacity.

At functional level, the order of service functions, e.g. Chain#1 {s1, s4, s6}, Chain#2{s4, s7}, is important, but very often which instance of the Service Function "s1" is selected for the Chain #1 is not. It is also possible that multiple instances of one service function can be reached by different network nodes. The actual instance selected for a service chain is called "Service Chain Instance Path".

There are various policies that could be employed to select instances for service chain instance path. Some Service Function Instances are visible to Service Chain Path. Sometimes a collection of service function instances can appear as one single entity to the Service Chain Path, leaving the instance selection to local nodes.

When there is change to the instances selected for a Service Chain Instance Path, either in-band or out-of-band messages can be sent to the SFF nodes to update the steering policies dynamically.

The downside with out-of-band messages is synchronization and race conditions. For a newly recognized flow, it is not scalable to expect the classifier node to queue the packets until the out-of-band notification is acknowledged by every Service Function Forwarder node. On the other hand, it is reasonable to use out-of-band messages to inform steering policies on SFF nodes if the steering policies can be pre-established before traffic arrives at the Classifier nodes, e.g. subscriber profile basis service chain instance path.

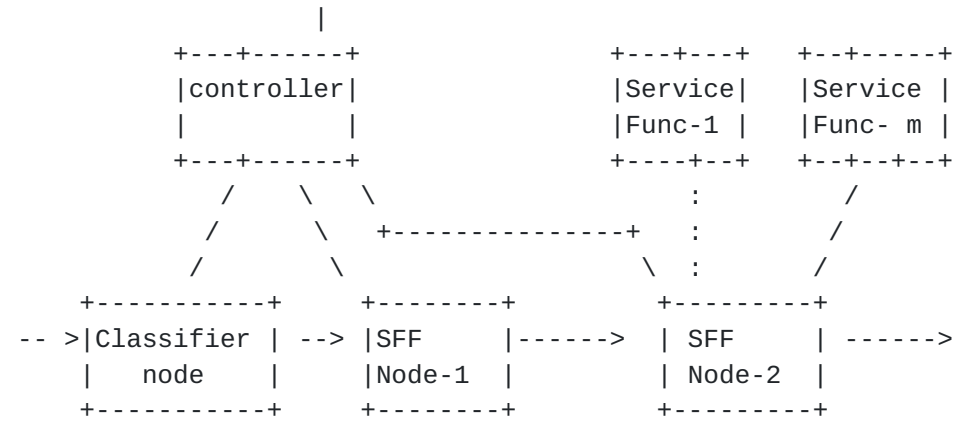

Figure 3 Controller for Service Chain Instance Path

Some service functions make changes to data packets, such as NAT changing the address fields. If any of those fields are used in traffic steering along the service chain, the criteria can be different before and after those the service functions.

5.2. Service Chain Re-Classification

The policy for associating flows with their service chains can be complicated and could be dynamic due to different behavior associated with chains.

For a chain of {FW, Header_enrichment, smart_node, Video_opt, Parental Control}, the video optimizer really needs to work on the response path. It may also use completely different encapsulation e.g. ICAP for example. There could be Smart-Node to further classify a particular part of the flow and bypass something, say the "video_opt". Therefore, the classification done by the service chain classification nodes at the network entrance can't completely dictate the exact sequence of service functions.

Basically, some service functions, especially Layer 7 service functions, can re-classify the service chain. So a chain could be constructed explicitly like below:

```
Classifier -> (SF-A) -> (SF-B) -> (SF-L7 Classifier) -- Chain -X
                                          |
                                          +--   Chain Y
```

Essentially SF-L7 is more like deep classification engine that might analyze individual http transaction and classify them differently. In reality SF-L7 can be a reverse proxy that is then capable of handling individual http transaction and select appropriate chain.

For Chain Re-classification, it is necessary to have message level coordination among those SFs and Service Chain Orchestration or/and Controller entities, as shown in the following figure:

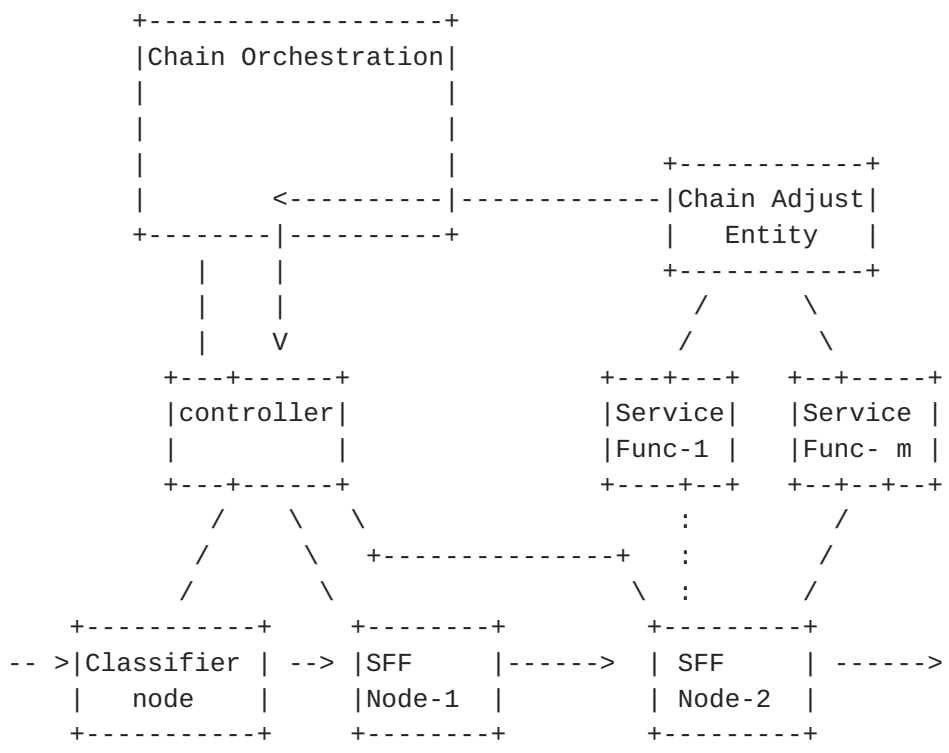

Figure 4 Service Chain Re-classification

The Service Chain Classification node can encounter flows that don't match with any policies. There should be a default policy that applies all statutorily required policies to the unknown flows.

Multiple flows can share one service chain. The criteria to select flows to be associated with their service chain could be different. For example, for one service chain "A" shared by Flow X, Y, Z:

- Criteria for Flow X to the Service Chain "A" are TCP port
- Criteria for Flow Y to the Service Chain "A" are Destination Address
- Criteria for Flow Z to the Service Chain "A" are MPLS label.

5.3. Layer 4-7 traffic Steering

Very often the criteria for steering flows to service functions are based on higher layer headers, such as TCP header, HTTP header, etc.

Most of deployed switches/routers are very efficient in forwarding packets based on Layer 2 or Layer 3 headers, such as MAC/IP destination addresses, or VLAN/MPLS labels but have limited capacity for forwarding data packets based on higher layer header. As of today, differentiating data packets based on higher layer headers depends on ACLs (Access Control List field matching) or DPI, both of which are relatively expensive and extensive use of such facilities may limit the bandwidth of switches/routers.

The Service Chain classification node introduced by [Boucadair-framework] and [SFC-ARCH] can alleviate the workload on large number of nodes in the network, including SFF nodes, from steering traffic based on higher layer fields.

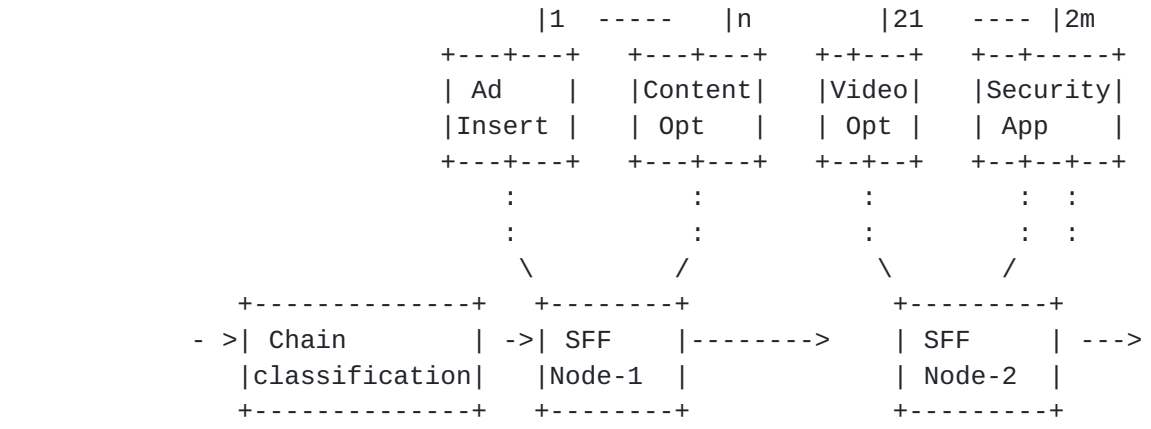

Figure 5 Service Chain Marking At Ingress

A Service Chain Classification node can associate a unique Service Chain Label (e.g. Layer 2 or 3 Label) or SF MAP Index to the packets in the flow. Such a Layer 2 or 3 Label makes it easier for subsequent nodes along the flow path to steer the flow to the service functions specified by the flow's service chain.

The Service Chain Classification Function usually resides on the ingress edge nodes of the service chain domain, such as Wireless Packet Gateway, Broadband Network Gateways, Cell Site Gateways, etc.

In some situations, like service chain for wireless subscribers, many flows (i.e. subscribers) have common service chain requirements. Under those situations, the Service Chain classification Functional can mark multiple flows with the same service chain requirement using the same Layer 2 or 3 Label, which effectively aggregates those flows into one service chain.

For service chains that are shared by a great number of flows, they can be pre-provisioned. For example, if VLAN ID=10 is the service chain that need to traverse "Service-1" at SFF Node #1 and "Service-3" at SFF Node #2, the steering policy for VLAN ID=10 can be dynamically changed by controllers.

6. Service Chain from the Layer 7 Perspective

From the Layer 7 perspective, the service chain can be much more complex. As shown in the figure below, the service functions to be chained can depend on the HTTP message request and reply. The service chain classification nodes may have to examine the whole HTTP message to determine the specific sequence of service functions for the flows. The HTTP message might have to be extracted from multiple data packets. Sometimes, the logic to steer traffic to chain of service functions might depend on the data retrieved from a database based on messages constructed from packets. The decision may depend on the HTTP response rather than the request, or it may depend on a particular sequence of request-response messages. The message handler may also alter the Layer 7 service chain based on hints or modification done by previous service function. HTTP based service function may insert HTTP header to add further criterion for service selection in the next round of classification.

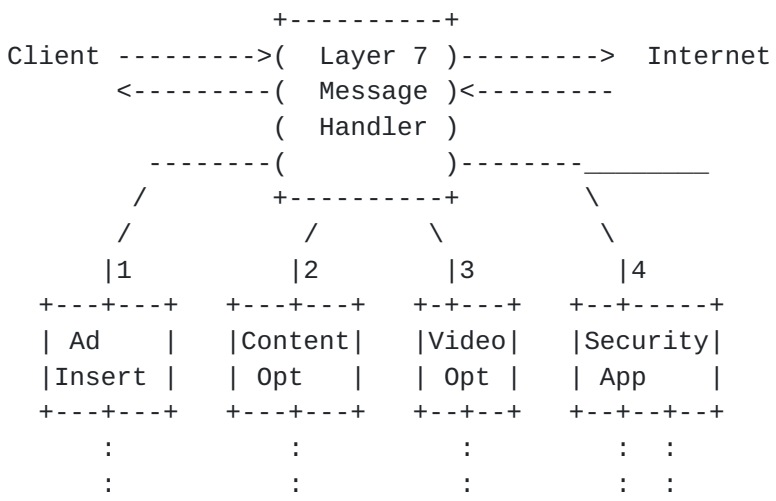

Figure 6 Layer 7 Service Chain Complexity

7. Conclusion and Recommendation

There are many Layer 4-7 service functions being deployed in the network. Many of them are not capable to adapt to new service chain encapsulation layer.

This document provides architecture framework for chaining those Layer 4-7 service functions that are not aware of new service layer encapsulation.

8. Manageability Considerations

There currently exists no single management methodology to control the L2-4 packet-based forwarding device, the L4-7 service delivery device, and the L7+ application server. Such unified management of configuration state is required for service function chaining to be a practical solution.

9. Security Considerations

TBD

10. IANA Considerations

This document requires no IANA actions. RFC Editor: Please remove this section before publication.

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

11.2. Informative References

[Boucadair-framework] M. Boucadair, et al, "Differentiated Service Function Chaining Framework", < draft-boucadair-service-chaining-framework-00>; Aug 2013

[SFC-Problem] P. Quinn, et al, "Service Function Chaining Problem statement", <draft-quinn-sfc-problem-statement-02>, Dec 9, 2013

[SFC-Framework] M. Boucadair, et al, "Service Function Chaining: Framework & Architecture", < draft-boucadair-sfc-framework-00>; Oct 2013

[SFC-Arch] J. Halpern, et al, "Service Function Chaining (SFC) Architecture", < draft-merged-sfc-architecture-00>, July 2014.

[NSH-Header] P. Quinn, et al, "Network Service Header", < draft-quinn-nsh-01>, July 12, 2013

[SC-MobileNetwork] W. Haeffner, N. Leymann, "Network Based Services in Mobile Network", IETF87 Berlin, July 29 2013

[Application-SDN] J. Giacomonni, "Application Layer SDN", Layer 123 ONF Presentation, Singapore, June 2013

[SC-Use-Case] Liu, et, al., "Service Chaining Use Cases", < draft-liu-service-chaining-use-cases-00>, Sept, 2013

12. Acknowledgments

This draft has merged some sections from http://datatracker.ietf.org/doc/draft-parker-sfc-chain-to-path/.

This draft has taken input from "Application Layer SDN" presentation given by John Giacomoni of F5 at Layer 123 conference. Thanks to Huang Shi Bi and Li Hong Yu for the valuable comments and suggestions.

This document was prepared using 2-Word-v2.0.template.dot.

Authors' Addresses

Linda Dunbar
Huawei Technologies
5340 Legacy Drive, Suite 175
Plano, TX 75024, USA
Phone: (469) 277 5840
Email: ldunbar@huawei.com

Ron Parker
Affirmed Networks
Acton, MA 01720
USA
Email: ron_parker@affirmednetworks.com

Ian Smith
F5 Networks
Email: I.Smith@F5.com

Sumandra Majee
F5 Networks
Email: S.Majee@F5.com

Ning So
Vinci Systems
Email: ning.so@vinci-systems.com

Donald Eastlake
Huawei Technologies
155 Beaver Street
Milford, MA 01757 USA
Phone: 1-508-333-2270
Email: d3e3e3@gmail.com