Service Function Chaining
Internet-Draft
Intended status: Standards Track
Expires: November 17, 2015

S. Kumar, Ed.
L. Kreeger, Ed.
Cisco Systems, Inc.
S. Majee
F5 Networks
W. Haeffner
Vodafone
R. Manur
Broadcom

May 16, 2015

# UDP Overlay Transport For Network Service Header
## draft-kumar-sfc-nsh-udp-transport-00

## Abstract

This draft describes the transport encapsulation to carry Network Service Header (NSH) over UDP protocol. This enables applications and services using NSH to communicate over a simple layer-3 network without topological constraints. It brings down the barrier to implement overlay transports by not requiring additional overhead as is typical of overlay mechanisms designed on top of UDP.

As a first benefit, this method eases the deployment of Service Function Chaining (SFC) by allowing SFC components to utilize the basic UDP/IP stack available in virtually all network elements and end systems to setup the overlays and realize SFCs.

Table of Contents



## 1. Introduction

NSH is an encapsulation designed to carry SFC specific information and metadata. It is very flexible in providing fixed and variable length encapsulation options while allowing for a high degree of extensibility. NSH in addition allows for carrying a variety of packets as payload, there by being just a shim header between the

inner payload and the outer transport.

NSH focuses on the application aspect of the encapsulation while leaving the transport mechanisms out of scope. This design choice





allows NSH to be carried on any overlay transport as required by the application and the use cases.

The transport independence aspect of NSH makes it necessary for existing transport protocols or new ones to carry NSH encapsulated packet as a payload. Given that IP networks are ubiquitous with virtually every device, element, node connected to the IP network possessing the ability to support UDP datagram transport over IP layer, it is one of the most basic of the transports to carry NSH.

UDP as a transport provides many benefits which has made it the de-facto choice for overlay networks such as VxLAN [RFC7348]. By nature it is a datagram service and trades reliability for simplicity and reduced overhead. It allows for sufficient entropy, for the network to exploit, in load balancing packets across paths in the network. Likewise, end hosts exploit it to distribute packets between the NICs and processor cores, within, for optimum performance. To this end, network elements and end hosts, both hardware and software, implement specific mechanisms to optimize UDP packet processing.

UDP datagram service and efficient implementations of it in existing networks is thus a forgone conclusion. These benefits among others, coupled with extensibility aspect of NSH - to implement security, header verification, etc., makes UDP a very simple, widely available and foundational choice for transporting NSH encapsulated packets.

This draft describes the creation of on-demand point-to-point lightweight NSH overlays using UDP as the overlay transport mechanism.

## 1.1. Requirements Language

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].

## 2. Definition Of Terms

This document uses some terms defined in SFC architecture [I-D.ietf-sfc-architecture] and NSH [I-D.ietf-sfc-nsh] drafts as mere examples for ease of understanding.





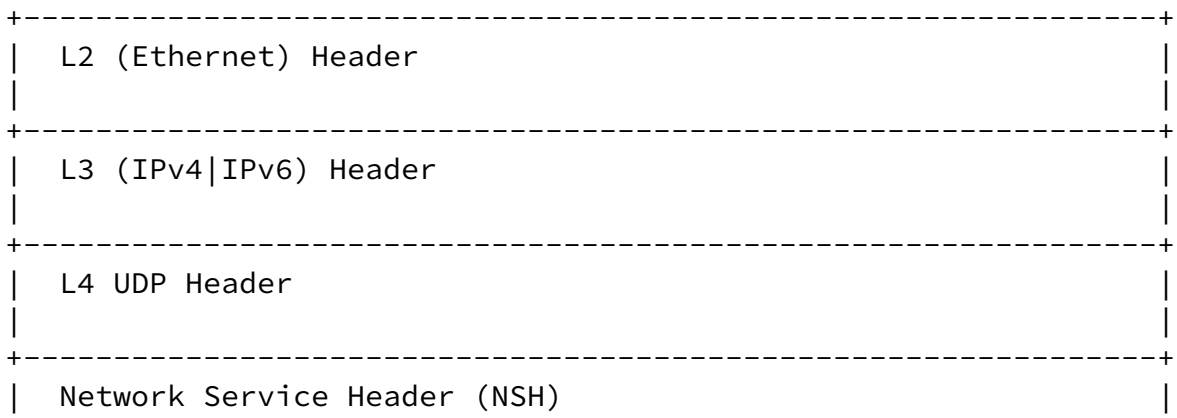
## 3. NSH UDP Overlay Transport

### 3.1. Stacking And Layering

A NSH encapsulated packet when carried over an UDP overlay transport looks as depicted in Figure 1.

The original payload, L2 frame, L3 packet, NSH OAM message, etc., is first encapsulated in NSH shim header. The NSH encapsulated packet then becomes the payload for the UDP packet carried over an IPv4 or IPv6 network. The UDP header serves as the L4 overlay transport for NSH and its payload.

Although depicted as a layer3 IP over an L2 network, nothing is assumed about how the L3 network is designed and deployed. It is entirely possible for IPinIP or MPLS or other underpinnings.



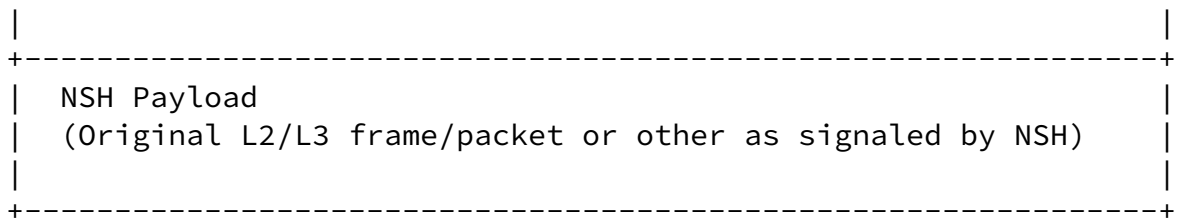



Figure 1: NSH UDP Stack

## 3.2. NSH UDP Overlay Packet Format

Figure 2 shows the format of the NSH encapsulation transported over UDP.





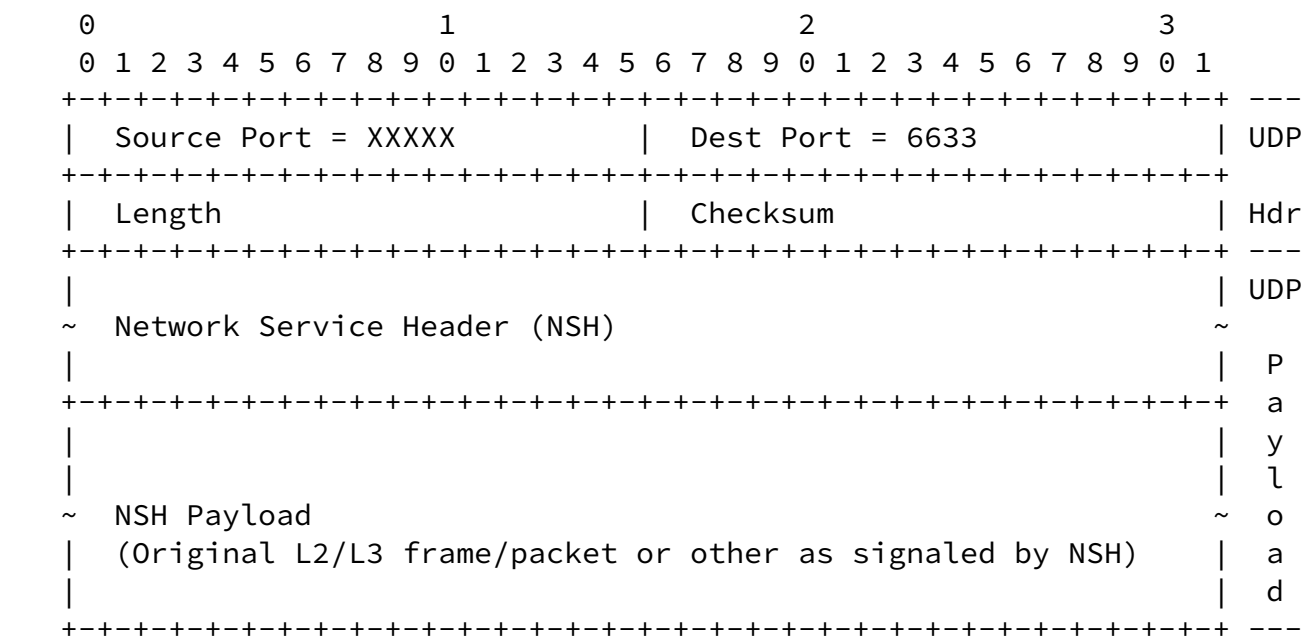



Figure 2: NSH UDP Overlay Encapsulation Format

Source Port :

The UDP port number computed to provide entropy. See Section 3.4 for details.

Dest Port :
UDP port number assigned to NSH: 6633.

Length :
Length of the UDP payload. This includes both the UDP header and payload.

Checksum :
Standard UDP checksum or zero.

NSH :
The NSH encapsulation.

NSH Payload :
The original frame or packet being carried or OAM message, etc.

## 3.3. Overlay Transport End-points

The UDP overlay transport extends between the two end-points involved in carrying the NSH overlay traffic. The control plane provisioning the NSH overlay MUST specify the location of the overlay destination when using UDP transport overlay, such as the IPv4 or IPv6 address of the end-point.





In the case of SFC, this UDP overlay transport extends between two SFC components: Classifier and SFF or SFFs or SFF and SF or SFF and SFC-proxy. The destination of the UDP overlay transport is thus the IP address used by these components to receive the NSH overlay traffic. When UDP overlay transport is required to carry NSH encapsulated traffic, SFC control plane MUST provision the UDP overlay transport destination and the use of UDP overlay transport.

## 3.4. UDP Source Port Considerations

The source port used in the UDP overlay transport SHOULD be computed to provide entropy for load balancing along the transmission path, including network elements such as routers and switches as well as end points such as servers. This behavior may in turn be controlled by local-policy at the encapsulating entity.

The source UDP port number SHOULD stay constant and not change for the flow represented within the NSH payload. This is typically done by computing the source UDP port number as a hash over the invariant part of the NSH payload. This could be IP and UDP or IP and TCP part of the NSH payload when the next-protocol field in NSH base header is set to IPv4, for instance. This avoids inducing packet reordering due to the use of NSH UDP overlay transport.

The recommended selection of source port as per [RFC6335], is the dynamic range: 49152-65535. A number in this range SHOULD be selected to reflect the NSH payload.

## 3.5. Checksum Considerations

The checksum in the UDP header MAY be set to zero for performance or other implementation specific reasons by the entity encapsulating the NSH packet (classifier, SFF, SF-proxy or SF). The receiving entity thus MUST accept a UDP encapsulated NSH packet with zero UDP checksum.

Implementations MAY choose to use non-zero checksum values. When a checksum other than zero is set by the encapsulating entity, it MUST be computed over the IP and UDP headers as defined in the UDP specification [RFC0768]. The receiving entity thus MUST accept a UDP encapsulated NSH packet with non-zero UDP checksum. Receiving entities, of NSH UDP overlay packets with non-zero checksum, are RECOMMENDED to verify the checksum before accepting the packet.





## 3.6. MTU Considerations

Operators of networks deploying UDP overlay transport for NSH are RECOMMENDED to configure the MTU of the network to accommodate NSH and UDP transport encapsulation overhead. This prevents fragmentation of UDP overlay transport encapsulated NSH packets and the overhead of processing such fragments both in the network and the end-points.

## 3.7. Fragmentation Considerations

Entities performing the UDP transport encapsulation MUST use the same source port number on all the fragments of the same packet when encapsulating pre-fragmented IP packets.

## 3.8. UDP-Lite Considerations

Exercising the option of setting the NSH UDP encapsulation checksum to zero, does not protect the NSH header from errors introduced into the header during transmission. NSH provides extensibility for applications or future NSH extensions to build such bit error protection.

Implementations that require protection against bit errors MAY use UDP-lite [RFC3828] with checksum coverage covering the NSH header. UDP-lite shares the UDP name space but uses the IP protocol identifier to distinguish itself from UDP.

# 4. IANA Considerations

IANA is requested to de-assign the well-known UDP port number 6633 and re-assign it for the purpose defined in this draft.

# 5. Security Considerations

Encapsulating NSH in UDP does not alter the security risk of NSH encapsulation and payload.

Security of the payload encapsulated by NSH is as defined in [I-D.ietf-sfc-nsh]

# 6. References

## 6.1. Normative References

[I-D.ietf-sfc-nsh]
Quinn, P. and U. Elzur, "Network Service Header", draft-ietf-sfc-nsh-00 (work in progress), March 2015.





[RFC0768] Postel, J., "User Datagram Protocol", STD 6, RFC 768,

August 1980.

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

[RFC3828] Larzon, L-A., Degermark, M., Pink, S., Jonsson, L-E., and G. Fairhurst, "The Lightweight User Datagram Protocol (UDP-Lite)", RFC 3828, July 2004.

## 6.2. Informative References

[I-D.ietf-sfc-architecture]
Halpern, J. and C. Pignataro, "Service Function Chaining (SFC) Architecture", draft-ietf-sfc-architecture-05 (work in progress), February 2015.

[RFC6335] Cotton, M., Eggert, L., Touch, J., Westerlund, M., and S. Cheshire, "Internet Assigned Numbers Authority (IANA) Procedures for the Management of the Service Name and Transport Protocol Port Number Registry", BCP 165, RFC 6335, August 2011.

[RFC7348] Mahalingam, M., Dutt, D., Duda, K., Agarwal, P., Kreeger, L., Sridhar, T., Bursell, M., and C. Wright, "Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", RFC 7348, August 2014.

## Authors' Addresses

Surendra Kumar (editor)
Cisco Systems, Inc.
170 W. Tasman Dr.
San Jose, CA 95134
US

Email: smkumar@cisco.com

Larry Kreeger (editor)
Cisco Systems, Inc.
170 W. Tasman Dr.
San Jose, CA 95134
US

Email: kreeger@cisco.com

Sumandra Majee
F5 Networks
90 Rio Robles
San Jose, CA 95134
US

Email: S.Majee@F5.com

Walter Haeffner
Vodafone
Ferdinand-Braun-Platz 1
Duesseldorf 40549
DE

Email: walter.haeffner@vodafone.com

Rajeev Manur
Broadcom

Email: rmanur@broadcom.com