

Network Working Group
Internet-Draft
Intended status: Informational
Expires: January 9, 2020

X. Geng
Z. Li
M. Chen
Huawei
July 8, 2019

SRv6 for Deterministic Networking (DetNet)
draft-geng-spring-srv6-for-detnet-00

Abstract

Deterministic Networking provides service with low jitter, bounded latency and low loss rate, using technologies of explicit route, resource reservation and service protection. This document specifies how to implement Deterministic Networking (DetNet) in a SRv6 Network.

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].

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 January 9, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Terminology	3
3. SRv6 for DetNet Overview	4
4. Service Protection	5
4.1. Service Protection Scenarios	6
4.2. Data Plane Considerations	8
4.3. Control Plane Considerations	8
4.4. Functions for Service Protection	9
4.4.1. End. B. Replication: Packet Replication Function	9
4.4.2. End. B. Elimination: Packet Elimination Function	9
5. Explicit Route	10
6. IANA Considerations	10
7. Security Considerations	10
8. Acknowledgements	10
9. Normative References	10
Authors' Addresses	11

1. Introduction

With more and more applications running in the Internet, quality of the service gains more and more attention, especially for some new applications, for example Cloud VR, Cloud Game, HDV (high-definition video) and so on, SLA (Service Level Agreement), including jitter, delay and loss rate, influences the users' experience significantly. So SLA guarantee is an important issue which requires new technologies and solutions for the network.

Deterministic Networking (DetNet defined in [I-D.ietf-detnet-architecture]) provides a capability to carry specified data flows for real-time applications with extremely low data loss rates, low jitter and bounded latency within a network domain. Techniques used include: 1) providing explicit paths for DetNet flows that satisfies the SLA requirement from user and do not immediately change with the network topology; 2) reserving data plane resources for DetNet flows along the allocated path of the flow; 3) replicates DetNet flows into two or more copies and transport different copies through different path in parallel, called service protection.

Segment Routing (SR) leverages the source routing paradigm. An ingress node steers a packet through an ordered list of instructions, called "segments". SR can be applied over IPv6 data plane using Routing Extension Header (SRH, [I-D.ietf-6man-segment-routing-header]). A segment in Segment Routing is not limited to a routing/forwarding function. An SRv6 Segment can indicate functions that are executed locally in the node where they are defined.

[I-D.filsfils-spring-srv6-network-programming] describes some well-known functions and segments associated to them. SRH TLVs ([I-D.ietf-6man-segment-routing-header]) also provides meta-data for segment processing. All these features make SRv6 suitable to carry DetNet flows, by defining new segments associated with DetNet functions and meta data for DetNet.

This document describes the concepts needed by implementing DetNet in an SRv6-based domain and provides considerations for any corresponding implementation.

Editor's note:

1. DetNet is limited in a controlled network domain, and it is not the only method to provide SLA guarantee. But it is a good start to discuss how to use SRv6 to have a SLA-guaranteed network service.

2. Resource Reservation will be added in future work.

2. Terminology

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 [RFC2119].

Terminologies for DetNet go along with the definition in [I-D.ietf-detnet-architecture] and [I-D.ietf-6man-segment-routing-header]:

DetNet domain

The portion of a network that is DetNet aware. It includes end systems and DetNet nodes

DetNet edge node

An instance of a DetNet relay node that acts as a source and/ or destination at the DetNet service sub-layer. For example, it can include a DetNet service sub-layer proxy function for DetNet service protection (e.g., the addition or removal of packet

sequencing information) for one or more end systems, or starts or terminates resource allocation at the DetNet forwarding sub-layer, or aggregates DetNet services into new DetNet flows. It is analogous to a Label Edge Router (LER) or a Provider Edge (PE) router.

DetNet relay node

A DetNet node including a service sub-layer function that interconnects different DetNet forwarding sub-layer paths to provide service protection. A DetNet relay node participates in the DetNet service sub-layer. It typically incorporates DetNet forwarding sub-layer functions as well, in which case it is collocated with a transit node.

DetNet transit node

A DetNet node operating at the DetNet forwarding sub-layer, that utilizes link layer and/or network layer switching across multiple links and/or sub-networks to provide paths for DetNet service sub-layer functions. Typically provides resource allocation over those paths. An MPLS LSR is an example of a DetNet transit node.

End system

End systems of interest to this document are either sources or destinations of DetNet flows. An end system may or may not be DetNet forwarding sub-layer aware or DetNet service sub-layer aware.

DetNet service sub-layer

DetNet functionality is divided into two sub-layers. One of them is the DetNet service sub-layer, at which a DetNet service, e.g., service protection is provided.

DetNet forwarding sub-layer

DetNet functionality is divided into two sub-layers. One of them is the DetNet forwarding sub-layer, which optionally provides resource allocation for DetNet flows over paths provided by the underlying network.

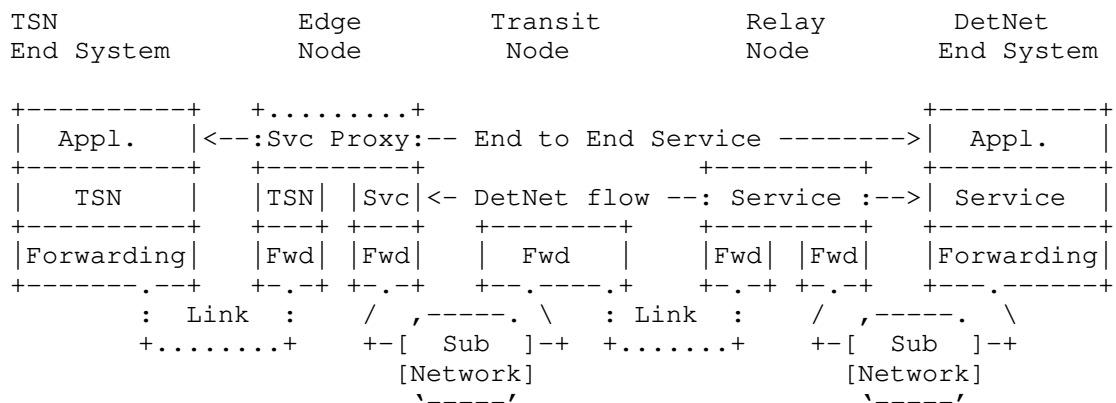
3. SRv6 for DetNet Overview

As mentioned above, there are mainly three technologies/functions defined in DetNet: Explicit Route, Resource Reservation and Service Protection. Explicit Route is the basis of the other two

technologies, and guarantees the path satisfies the SLA requirement from application. Resource Reservation protects DetNet flows from network congestion, which could reduce the end-to-end latency and congestion loss; Service Protection prevents DetNet flow from random media errors and equipment failures, which makes the loss rate extremely low.

In [I-D.ietf-detnet-architecture], DetNet functionality is implemented in two sub-layers in the protocol stack: the DetNet service sub-layer and the DetNet forwarding sub-layer. The DetNet service sub-layer provides DetNet service, e.g., service protection. The DetNet forwarding sub-layer supports DetNet service in the underlying network, by providing explicit routes and resource allocation to DetNet flows. There is no obvious protocol stack as MPLS, in SRv6 both service sub-layer and forwarding sub-layer are implemented through SRH.

The following picture shows that a general DetNet enabled network defined in [I-D.ietf-detnet-architecture] :

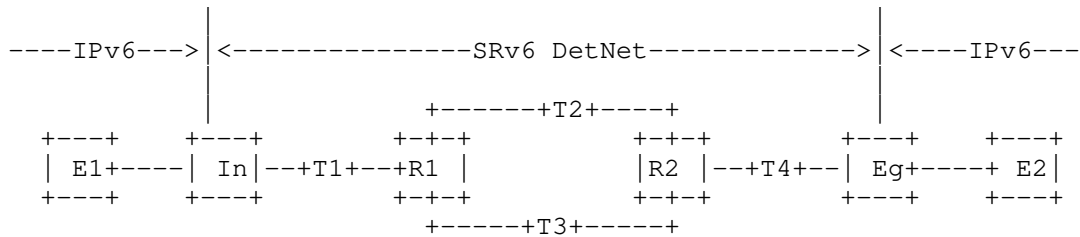


In SRv6 for DetNet, the outer IPv6 Header with the SRH is used for carrying DetNet flows. Explicit path is instantiated in the segment list of SRH, and other functions/arguments for service protection (packet replication, elimination and ordering, PREOF) and resource reservation (packet queuing and scheduling) are also defined in the specified SID. Meta data for DetNet could be instantiated in the Optional TLVs.

4. Service Protection

4.1. Service Protection Scenarios

The figure below shows that an IPv6 flow is sent out from the end station E1. The packet of the flow is encapsulated in an outer IPv6+SRH header as a DetNet SRv6 packet in the Ingress(In) and transported through an SRv6 DetNet domain. In the Egress(Eg), the outer IPv6 header+SRH of the packet is popped, and the packet is sent to the destination E2.



The DetNet packet processing is as follows:

Ingress:

Inserts the SRv6 Policy that will steer the packet from Ingress to the destination

The methods and mechanisms used for defining, instantiating and applying the policy are outside of this document. An example of policies are described in [I-D.ietf-spring-segment-routing-policy]

Flow Identification and Sequence Number are carried in the SRH.

Relay Node 1(Replication Node):

Replicates the payload and IPv6 Header with the SRH. This is a new function in the context of SRv6 Network Programming which will associate a given SID to a replication instruction in the node originating and advertising the SID. The replication instruction includes:

- * The removal of the existing IPv6+SRH header
- * The encapsulation into a new outer IPv6+SRH header. Each packet (the original and the duplicated) are encapsulated into respectively new outer IPv6+SRH headers.

Binding two different SRv6 Policies respectively to the original packet and the replicated packet, which can steer the packets from Relay Node 1 to Relay Node 2 through two tunnels.

Relay Node 2 (Elimination Node):

Eliminates the redundant packets.

Binds a new SRv6 Policy to the survival packet, which steers the packet from Relay Node 2 to Egress.

Egress:

Decapsulates the outer Ipv6 header.

Sends the inter packet to the End Station 2.

The DetNet packet encapsulation is illustrated here below. It has to be noted that, in the example below, the R2 address is a SRH SID associated to a TBD function related to the packet replication the node R1 has to perform. The same (or reverse) apply to node R2 which is in charge of the discard of the duplicated packet. Here also a new function will have a new SID allocated to it and representing the delete of the duplication in R2.

End Station1 output packet: (E1,E2)

Ingress output packet: (In, T1) (R1,T1, SL=2) (E1,E2)

Transit Node1 output packet: (In, R1) (R1,T1,SL=1) (E1,E2)

Relay Node1 output packets : (R1,T2) (R2,T2,SL=2) (E1,E2),
(R1,T3) (R2,T3,SL=2) (E1,E2)

Transit Node2 output packet: (R1, R2) (R2,T2,SL=1) (E1,E2)

Transit Node3 output packet: (R1, R2) (R2,T3,SL=1) (E1,E2)

Relay Node2 output packet: (R2, T4) (Eg,T4,SL=2) (E1,E2)

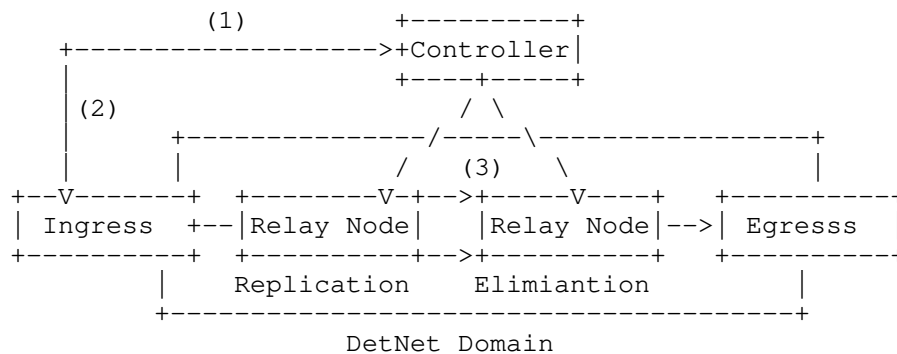
Transit Node4 output packet: (R2, Eg) (Eg,T4,SL=1) (E1,E2)

Egress out : (E1,E2)

4.2. Data Plane Considerations

Flow Identification and sequence number are necessary in the encapsulation of SRv6 for DetNet in order to support service protection. Replication nodes decide which DetNet flows are supposed to be replicated by identifying the flow with the flow identification. Elimination nodes decide whether a packet should be dropped because of redundancy by identifying the flow identification and sequence number.

4.3. Control Plane Considerations



1. Edge node->Controller: Sends a path computation requirement containing that service protection in order to have ultra-reliability through PCEP/BGP extensions.

2. Controller->Edge node: Computes a P2MP2P path, including replication nodes and elimination nodes. Between a pair of replication node and elimination node, there are more than one path, and if any equipment failure happens in one of them, the DetNet service is not interrupted. Send the path computation result to the edge node through PCEP/BGP extensions.

3. Controller->Relay Node : In a P2MP2P path, every pair of replication node and elimination node should be configured to identify different DetNet flows by the different flow identifications, and the rule of sequence number should be negotiated between relay nodes. After replication or elimination, the explicit path to the next relay is also required through BGP extensions or Netconf YANG.

4.4. Functions for Service Protection

New SRv6 Network Programming functions are defined as follows:

4.4.1. End. B. Replication: Packet Replication Function

1. IF NH=SRH & SL>0 THEN
2. extract the DetNet TLV values from the SRH
3. create two new outer IPv6+SRH headers: IPv6-SRH-1 and IPv6-SRH-2
Insert the policy-instructed segment lists in each newly created SRH (SRH-1 and SRH-2). Also, add the extracted DetNet TLVs into SRH-1 and SRH-2.
4. remove the incoming outer IPv6+SRH header.
5. create a duplication of the incoming packet.
6. encapsulate the original packet into the first outer IPv6+SRH header: (IPv6-SRH-1) (original packet)
7. encapsulate the duplicate packet into the second outer IPv6+SRH header: (IPv6-SRH-2) (duplicate packet)
8. set the IPv6 SA as the local address of this node.
9. set the IPv6 DA of IPv6-SRH-1 to the first segment of the SRv6 Policy in of SRH-1 segment list.
10. set the IPv6 DA of IPv6-SRH-2 to the first segment of the SRv6 Policy in of SRH-2 segment list.
11. ELSE
12. drop the packet

4.4.2. End. B. Elimination: Packet Elimination Function

1. IF NH=SRH & SL>0 & "the packet is not a redundant packet" THEN
2. do not decrement SL nor update the IPv6 DA with SRH[SL]
3. extract the value of DetNet TLVs from the SRH
4. create a new outer IPv6+SRH header

5. insert the policy-instructed segment lists in the newly created SRH and add the retrieved DetNet TLVs in the newly created SRH
6. remove the incoming outer IPv6+SRH header.
7. set the IPv6 DA to the first segment of the SRv6 Policy in the newly created SRH
8. ELSE
9. drop the packet

5. Explicit Route

SRv6 could support explicit route using segment list without extra extension. In DetNet, explicit route could be used with service protection or resource reservation. When explicit route works with service protection, a P2MP2P path is required to indicate the behavior of replication and elimination. When explicit route works with resource reservation, the explicit path indicates the nodes or links a DetNet flow goes through, and also indicates the resource allocated for the DetNet flow in the specified nodes or links.

6. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

7. Security Considerations

TBD

8. Acknowledgements

Thank you for valuable comments from James Guichard and Andrew Mails.

9. Normative References

[I-D.filsfils-spring-srv6-network-programming]
Filsfils, C., Camarillo, P., Leddy, J.,
daniel.voyer@bell.ca, d., Matsushima, S., and Z. Li, "SRv6
Network Programming", draft-filsfils-spring-srv6-network-
programming-07 (work in progress), February 2019.

- [I-D.ietf-6man-segment-routing-header]
Filsfils, C., Dukes, D., Previdi, S., Leddy, J.,
Matsushima, S., and d. daniel.voyer@bell.ca, "IPv6 Segment
Routing Header (SRH)", draft-ietf-6man-segment-routing-
header-21 (work in progress), June 2019.
- [I-D.ietf-detnet-architecture]
Finn, N., Thubert, P., Varga, B., and J. Farkas,
"Deterministic Networking Architecture", draft-ietf-
detnet-architecture-13 (work in progress), May 2019.
- [I-D.ietf-detnet-dp-sol-mpls]
Korhonen, J. and B. Varga, "DetNet MPLS Data Plane
Encapsulation", draft-ietf-detnet-dp-sol-mpls-02 (work in
progress), March 2019.
- [I-D.ietf-spring-segment-routing-policy]
Filsfils, C., Sivabalan, S., daniel.voyer@bell.ca, d.,
bogdanov@google.com, b., and P. Mattes, "Segment Routing
Policy Architecture", draft-ietf-spring-segment-routing-
policy-03 (work in progress), May 2019.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
Requirement Levels", BCP 14, RFC 2119,
DOI 10.17487/RFC2119, March 1997,
<<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L.,
Decraene, B., Litkowski, S., and R. Shakir, "Segment
Routing Architecture", RFC 8402, DOI 10.17487/RFC8402,
July 2018, <<https://www.rfc-editor.org/info/rfc8402>>.

Authors' Addresses

Xuesong Geng
Huawei

Email: gengxuesong@huawei.com

Zhenbin Li
Huawei

Email: lizhenbin@huawei.com

Mach Chen
Huawei

Email: mach.chen@huawei.com

DetNet
Internet-Draft
Intended status: Informational
Expires: April 29, 2020

B. Varga, Ed.
J. Farkas
Ericsson
L. Berger
D. Fedyk
LabN Consulting, L.L.C.
A. Malis
Independent
S. Bryant
Futurewei Technologies
J. Korhonen
October 27, 2019

DetNet Data Plane Framework
draft-ietf-detnet-data-plane-framework-03

Abstract

This document provides an overall framework for the DetNet data plane. It covers concepts and considerations that are generally common to any Deterministic Networking data plane specification.

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 April 29, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Terminology	4
2.1. Terms Used in This Document	4
2.2. Abbreviations	4
3. DetNet Data Plane Overview	5
3.1. Data Plane Characteristics	6
3.1.1. Data Plane Technology	6
3.1.2. Data Plane Format	6
3.2. Encapsulation	6
3.3. DetNet Specific Metadata	7
3.4. DetNet IP Data Plane	8
3.5. DetNet MPLS Data Plane	9
3.6. Further DetNet Data Plane Considerations	9
3.6.1. Per Flow Related Functions	9
3.6.2. Service Protection	11
3.6.3. Aggregation Considerations	13
3.6.4. End-System-Specific Considerations	14
3.6.5. Sub-Network Considerations	15
4. Controller Plane (Management and Control) Considerations	16
4.1. DetNet Controller Plane Requirements	16
4.2. Generic Controller Plane Considerations	17
4.2.1. Flow Aggregation Control	18
4.2.2. Explicit Routes	19
4.2.3. Contention Loss and Jitter Reduction	19
4.2.4. Bidirectional Traffic	20
4.3. Packet Replication, Elimination, and Ordering (PREOF)	21
5. Security Considerations	21
6. IANA Considerations	22
7. Acknowledgements	22
8. References	22
8.1. Normative References	22
8.2. Informative References	22
Authors' Addresses	25

1. Introduction

DetNet (Deterministic Networking) provides a capability to carry specified unicast or multicast data flows for real-time applications with extremely low packet loss rates and assured maximum end-to-end delivery latency. A description of the general background and concepts of DetNet can be found in [I-D.ietf-detnet-architecture].

This document describes the concepts needed by any DetNet data plane specification and provides considerations for any corresponding implementation. It covers the building blocks that provide the DetNet service, the DetNet service sub-layer and the DetNet forwarding sub-layer functions as described in the DetNet Architecture.

The DetNet Architecture models the DetNet related data plane functions decomposed into two sub-layers: a service sub-layer and a forwarding sub-layer. The service sub-layer is used to provide DetNet service protection and reordering. The forwarding sub-layer leverages Traffic Engineering mechanisms and provides congestion protection (low loss, assured latency, and limited out-of-order delivery).

As part of the service sub-layer functions, this document describes typical DetNet node data plane operation. It describes the function and operation of the Packet Replication (PRF) Packet Elimination (PEF) and the Packet Ordering (POF) functions within the service sub-layer. Furthermore, it also describes the forwarding sub-layer.

DetNet flows may be carried over network technologies that can provide the DetNet required service characteristics. For example, DetNet MPLS flows can be carried over IEEE 802.1 Time Sensitive Network (TSN) [IEEE802.1TSNTG] sub-networks. However, IEEE 802.1 TSN support is not required and some of the DetNet benefits can be gained by running over a data link layer that has not been specifically enhanced to support TSN.

Different application flows (e.g., Ethernet, IP, etc.), can be mapped on top of DetNet. DetNet can optionally reuse header information provided by, or shared with, applications. An example of shared header fields can be found in [I-D.ietf-detnet-ip].

This document also covers basic concepts related to the controller plane and Operations, Administration, and Maintenance (OAM). Data plane OAM specifics are out of scope for this document.

2. Terminology

2.1. Terms Used in This Document

This document uses the terminology established in the DetNet architecture [I-D.ietf-detnet-architecture], and the reader is assumed to be familiar with that document and its terminology.

2.2. Abbreviations

The following abbreviations are used in this document:

CW	Control Word.
d-CW	DetNet Control Word.
DetNet	Deterministic Networking.
DN	DetNet.
GRE	Generic Routing Encapsulation.
IPSec	IP Security.
L2	Layer 2.
LSR	Label Switching Router.
MPLS	Multiprotocol Label Switching.
MPLS-TE	Multiprotocol Label Switching - Traffic Engineering.
OAM	Operations, Administration, and Maintenance.
PEF	Packet Elimination Function.
PRF	Packet Replication Function.
PREOF	Packet Replication, Elimination and Ordering Functions.
POF	Packet Ordering Function.
PSN	Packet Switched Network.
PW	PseudoWire.
QoS	Quality of Service.

S-Label	DetNet "service" label.
TDM	Time-Division Multiplexing.
TSN	Time-Sensitive Network.

3. DetNet Data Plane Overview

This document describes how application flows, or app-flows, are carried over DetNet networks. The DetNet Architecture, [I-D.ietf-detnet-architecture], models the DetNet related data plane functions as decomposed into two sub-layers: a service sub-layer and a forwarding sub-layer.

Figure 1 reproduced from the [I-D.ietf-detnet-architecture], shows a logical DetNet service with the two sub-layers.

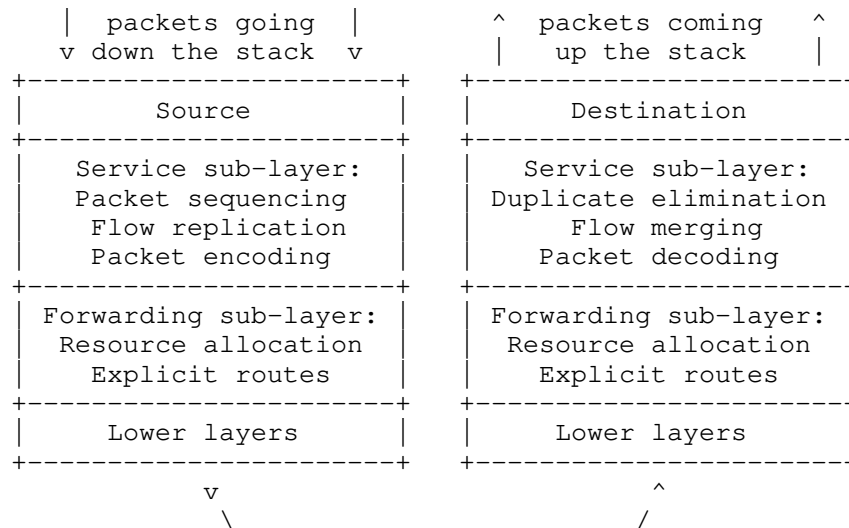


Figure 1: DetNet data plane protocol stack

The DetNet forwarding sub-layer may be directly provided by the DetNet service sub-layer, for example by IP tunnels or MPLS. Alternatively, an overlay approach may be used in which the packet is natively carried between key nodes within the DetNet network (say between PREOF nodes) and a sub-layer is used to provide the information needed to reach the next hop in the overlay.

The forwarding sub-layer provides the QoS related functions needed by the DetNet flow. It may do this directly through the use of queuing techniques and traffic engineering methods, or it may do this through

the assistance of its underlying connectivity. For example it may call upon Ethernet TSN capabilities defined in IEEE 802.1 TSN [IEEE802.1TSNTG].

The service sub-layer provides additional support beyond the connectivity function of the forwarding sub-layer. An example of this is Packet Replication, Elimination, and Ordering functions see Section 4.3.

The method of instantiating each of the layers is specific to the particular DetNet data plane method, and more than one approach may be applicable to a given bearer network type.

3.1. Data Plane Characteristics

There are two major characteristics to the data plane: the technology and the encapsulation, as discussed below.

3.1.1. Data Plane Technology

The DetNet data plane is provided by the DetNet service and forwarding sub layers. The DetNet service sub-layer generally provides its functions for the DetNet application flows by using or applying existing standardized headers and/or encapsulations. The Detnet forwarding sub-layer may provide capabilities leveraging that same header or encapsulation technology (e.g., DN IP or DN MPLS) or it may be achieved by other technologies (e.g., Figure 2). DetNet is currently defined for operation over packet switched (IP) networks or label switched (MPLS) networks.

3.1.2. Data Plane Format

DetNet encodes specific flow attributes (flow identity and sequence number) in packets. For example, in DetNet IP, zero encapsulation is used and no sequence number is available, and in DetNet MPLS, DetNet specific information may be added explicitly to the packets in the format of S-label and d-CW.

3.2. Encapsulation

The encapsulation of a DetNet flow allows it to be sent over a data plane technology other than its native type. DetNet uses header information to perform traffic classification, i.e., identify DetNet flows, and provide DetNet service and forwarding functions. As mentioned above, DetNet may add headers, as is the case for DN MPLS, or may use headers that are already present, as is the case in DN IP. Figure 2 illustrates some relationships between the components.

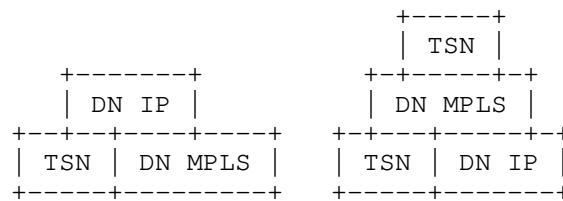


Figure 2: DetNet Service Examples

The use of encapsulation is also required if additional information (metadata) is needed by the DetNet data plane and there is either no ability to include it in the client data packet, or the specification of the client data plane does not permit the modification of the packet to include additional data. An example of such metadata is the inclusion of a sequence number required by the PREOF function.

Encapsulation may also be used to carry or aggregate flows for equipment with limited DetNet capability.

3.3. DetNet Specific Metadata

The DetNet data plane can provide or carry metadata:

1. Flow-ID
2. Sequence Number

The DetNet data plane framework supports a Flow-ID (for identification of the flow or aggregate flow) and/or a Sequence Number (for PREOF) for each DetNet flow. The DetNet Service sub-layer requires both; the DetNet forwarding sub-layer requires only Flow-ID. Metadata can also be used for OAM indications and instrumentation of DetNet data plane operation.

Metadata can be included implicit or explicit. Explicit means that a dedicated header field is used to include metadata in a DetNet packet. In case of implicit method a part of an already existing header field is used to encode the metadata.

Explicit inclusion of metadata is possible through the use of IP options or IP extension headers. New IP options are almost impossible to get standardized or to deploy in an operational network and will not be discussed further in this text. IPv6 extensions headers are finding popularity in current IPv6 development work, particularly in connection with Segment Routing of IPv6 (SRv6) and IP OAM. The design of a new IPv6 extension header or the modification

of an existing one is a technique available in the tool box of the DetNet IP data plane designer.

Explicit inclusion of metadata in an IP packet is also possible through the inclusion of an MPLS label stack and the MPLS DetNet Control Word using one of the methods for carrying MPLS over IP [I-D.ietf-detnet-mpls-over-udp-ip]. This is described in more detail in Section 3.6.5.

Implicit metadata in IP can be included through the use of the network programming paradigm [I-D.ietf-spring-srv6-network-programming] in which the suffix of an IPv6 address is used to encode additional information for use by the network of the receiving host.

Some MPLS examples of implicit metadata include the sequence number for use by the PREOF function, or even all the essential information being included into the DetNet over MPLS label stack (the DetNet Control Word and the DetNet Service label).

3.4. DetNet IP Data Plane

An IP data plane may operate natively or through the use of an encapsulation. Many types of IP encapsulation can satisfy DetNet requirements and it is anticipated that more than one encapsulation may be deployed, for example GRE, IPSec etc.

One method of operating an IP DetNet data plane without encapsulation is to use "6-tuple" based flow identification, where "6-tuple" refers to information carried in IP and higher layer protocol headers. General background on the use of IP headers, and "6-tuples", to identify flows and support Quality of Service (QoS) can be found in [RFC3670]. [RFC7657] provides useful background on differentiated services (DiffServ) and "tuple" based flow identification. DetNet flow aggregation may be enabled via the use of wildcards, masks, prefixes and ranges. The operation of this method is described in detail in [I-D.ietf-detnet-ip].

The DetNet forwarding plane may use explicit route capabilities and traffic engineering capabilities to provide a forwarding sub-layer that is responsible for providing resource allocation and explicit routes. It is possible to include such information in a native IP packet explicitly, or implicitly.

3.5. DetNet MPLS Data Plane

MPLS provides the ability to forward traffic over implicit and explicit paths to the point in the network where the next DetNet service sub-layer action needs to take place. It does this through the use of a stack of one or more labels with various forwarding semantics.

MPLS also provides the ability to identify a service instance that is used to process the packet through the use of a label that maps the packet to a service instance.

In cases where metadata is needed to process an MPLS encapsulated packet at the service sub-layer, a shim layer called a control word (CW) [RFC4385] can be used. Although such CWs are frequently 32 bits long, there is no architectural constraint on its size of this structure, only the requirement that it is fully understood by all parties operating on it in the DetNet service sub-layer. The operation of this method is described in detail in [I-D.ietf-detnet-mpls].

3.6. Further DetNet Data Plane Considerations

This section provides informative considerations related to providing DetNet service to flows which are identified based on their header information.

3.6.1. Per Flow Related Functions

At a high level, the following functions are provided on a per flow basis.

3.6.1.1. Reservation and Allocation of resources

Reservation of resources can allocate resources to specific DetNet flows. This can eliminate packet contention and packet loss for DetNet traffic. This also can reduce jitter for DetNet traffic. Resources allocated to a DetNet flow protect it from other traffic flows. On the other hand, DetNet flows are assumed to behave with respect to the reserved traffic profile. Misbehaving DetNet flows must be detected and it have to be ensured that they do not compromise QoS of other flows. The use of (queuing, policing, shaping) policies can be used to ensure that the allocation of resources reserved for DetNet is met.

3.6.1.2. Explicit routes

Use of a specific path for a flow. This allows control of the network delay by steering the packet with the ability to influence the physical path. Explicit routes complement reservation by ensuring that a consistent path can be associated with its resources for the duration of that path. Coupled with the traffic mechanism, this limits misordering and bounds latency. Explicit route computation can encompass a wide set of constraints and optimize the path for a certain characteristic e.g. highest bandwidth or lowest jitter. In these cases the "best" path for any set of characteristics may not be a shortest path. The selection of path can take into account multiple network metrics. Some of these metrics are measured and distributed by the routing system as traffic engineering metrics.

3.6.1.3. Service protection

Use of multiple packet streams using multiple paths, for example 1+1 or 1:1 linear protection. For DetNet this primarily relates to packet replication and elimination capabilities. MPLS offers a number of protection schemes. MPLS hitless protection can be used to switch traffic to an already established path in order to restore delivery rapidly after a failure. Path changes, even in the case of failure recovery, can lead to the out of order delivery of data requiring packet ordering functions either within the DetNet service or at a high layer in the application traffic. Establishment of new paths after a failure is out of scope for DetNet services.

3.6.1.4. Network Coding

Network Coding, not to be confused with network programming, comprises several techniques where multiple data flows are encoded. These resulting flows can then be sent on different paths. The encoding operation can combine flows and error recovery information. When the encoded flows are decoded and recombined the original flows can be recovered. Note that Network coding uses an alternative to packet by packet PREOF. Therefore, for certain network topologies and traffic loads, Network Coding can be used to improve a network's throughput, efficiency, latency, and scalability, as well as resilience to partition, attacks, and eavesdropping, as compared to traditional methods. DetNet could utilize Network coding as an alternative to other protection means. Network coding is often applied in wireless networks and is being explored for other network types.

3.6.1.5. Load sharing

Use of packet-by-packet distribution of the same DetNet flow over multiple paths is not recommended except for the cases listed above where PREOF is utilized to improve protection of traffic and maintain order. Packet by packet load sharing, e.g., via ECMP or UCMP, impacts ordering and possibly jitter.

3.6.1.6. Troubleshooting

Detnet leverages many different forwarding sub-layers, each of which supports various tools to troubleshoot connectivity, for example identification of misbehaving flows. The DetNet Service layer can leverage existing mechanisms to troubleshoot or monitor flows, such as those in use by IP and MPLS networks. At the Application layer a client of a DetNet service can use existing techniques to detect and monitor delay and loss.

3.6.1.7. Flow recognition for analytics

Network analytics can be inherited from the technologies of the Service and Forwarding sub-layers. At the DetNet service edge, packet and bit counters (e.g. sent, received, dropped, and out-of-sequence) can be maintained.

3.6.1.8. Correlation of events with flows

The provider of a DetNet service may provide other capabilities to monitor flows, such as more detailed loss statistics and time stamping of events. The details of these capabilities are currently out of scope for this document.

3.6.2. Service Protection

Service protection allow DetNet services to increase reliability and maintain a DetNet Service Assurance in the case of network congestion or network failure. Detnet relies on the underlying technology capabilities for various protection schemes. Protection schemes enable partial or complete coverage of the network paths and active protection with combinations of PRF, PEF, and POF.

3.6.2.1. Linear Service Protection

An example DetNet MPLS network fragment and packet flow is illustrated in Figure 3.

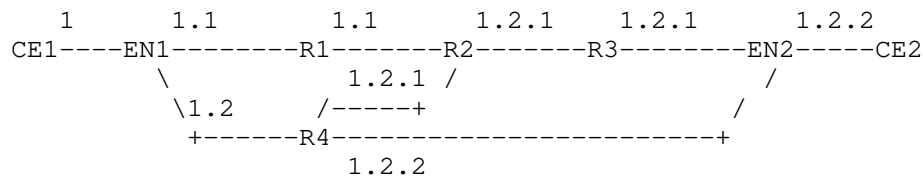


Figure 3: Example Packet Flow in DetNet protected Network

In Figure 3 the numbers are used to identify the instance of a packet. Packet 1 is the original packet, and packets 1.1, and 1.2 are two first generation copies of packet 1. Packet 1.2.1 is a second generation copy of packet 1.2 etc. Note that these numbers never appear in the packet, and are not to be confused with sequence numbers, labels or any other identifier that appears in the packet. They simply indicate the generation number of the original packet so that its passage through the network fragment can be identified to the reader.

Customer Equipment CE1 sends a packet into the DetNet enabled network. This is packet (1). Edge Node EN1 encapsulates the packet as a DetNet Packet and sends it to Relay node R1 (packet 1.1). EN1 makes a copy of the packet (1.2), encapsulates it and sends this copy to Relay node R4.

Note that along the path from EN1 to R1 there may be zero or more nodes which, for clarity, are not shown. The same is true for any other path between two DetNet entities shown in Figure 3 .

Relay node R4 has been configured to send one copy of the packet to Relay Node R2 (packet 1.2.1) and one copy to Edge Node EN2 (packet 1.2.2).

R2 receives packet copy 1.2.1 before packet copy 1.1 arrives, and, having been configured to perform packet elimination on this DetNet flow, forwards packet 1.2.1 to Relay Node R3. Packet copy 1.1 is of no further use and so is discarded by R2.

Edge Node EN2 receives packet copy 1.2.2 from R4 before it receives packet copy 1.2.1 from R2 via relay Node R3. EN2 therefore strips any DetNet encapsulation from packet copy 1.2.2 and forwards the packet to CE2. When EN2 receives the later packet copy 1.2.1 this is discarded.

The above is of course illustrative of many network scenarios that can be configured.

This example also illustrates 1:1 protection scheme meaning there is traffic over each segment of the end to end path. Local DetNet relay nodes determine which packets are eliminated and which packets are forwarded. A 1+1 scheme where only one path is used for traffic at a time, could use the same topology. In this case there is no PRF function and traffic is switched upon detection of failure. An OAM scheme that monitors the paths detects the loss of path or traffic is required to initiate the switch. A POF may still be used in this case to prevent misordering of packets. In both cases the protection paths are established and maintained for the duration of the DetNet service.

3.6.2.2. Ring Service Protection

Ring protection may also be supported if the underlying technology supports it. Many of the same concepts apply however rings are normally 1+1 protection for data efficiency reasons. [RFC8227] is an example of MPLS-TP data plane that supports Ring protection.

3.6.3. Aggregation Considerations

The DetNet data plane also allows for the aggregation of DetNet flows, which can improve scalability by reducing the per-hop state. How this is accomplished is data plane or control plane dependent. When DetNet flows are aggregated, transit nodes provide service to the aggregate and not on a per-DetNet flow basis. When aggregating DetNet flows the flows should be compatible i.e. the same or very similar QoS and CoS characteristics. In this case, nodes performing aggregation will ensure that per-flow service requirements are achieved.

If bandwidth reservations are used, the sum of the reservations should be the sum of all the individual reservations; in other words, the reservations should not add up to an over-subscription of bandwidth reservation. If maximum delay bounds are used, the system should ensure that the aggregate does not exceed the delay bounds of the individual flows.

When an encapsulation is used the choice of reserving a maximum resource level and then tracking the services in the aggregated service or adjusting the aggregated resources as the services are added is implementation and technology specific.

DetNet flows at edges must be able to handle rejection to an aggregation group due to lack of resources as well as conditions where requirements are not satisfied.

3.6.3.1. IP Aggregation

IP aggregation has both data plane and controller plane aspects. For the data plane, flows may be aggregated for treatment based on shared characteristics such as 6-tuple. Alternatively, an IP encapsulation may be used to tunnel an aggregate number of DetNet Flows between relay nodes.

3.6.3.2. MPLS Aggregation

MPLS aggregation also has data plane and controller plane aspects. MPLS flows are often tunneled in a forwarding sub-layer, under the reservation associated with that MPLS tunnel.

3.6.4. End-System-Specific Considerations

Data-flows requiring DetNet service are generated and terminated on end-systems. Encapsulation depends on the application and its preferences. For example, in a DetNet MPLS domain the sub-layer functions use the d-CWs, S-Labels and F-Labels to provide DetNet services. However, an application may exchange further flow related parameters (e.g., time-stamp), which are not provided by DetNet functions.

As a general rule, DetNet domains are capable of forwarding any DetNet flows and the DetNet domain does not mandate the end-system or edge node encapsulation format. Unless there is a proxy of some form present, end-systems peer with similar end-systems using the same application encapsulation format. For example, as shown in Figure 4, IP applications peer with IP applications and Ethernet applications peer with Ethernet applications.

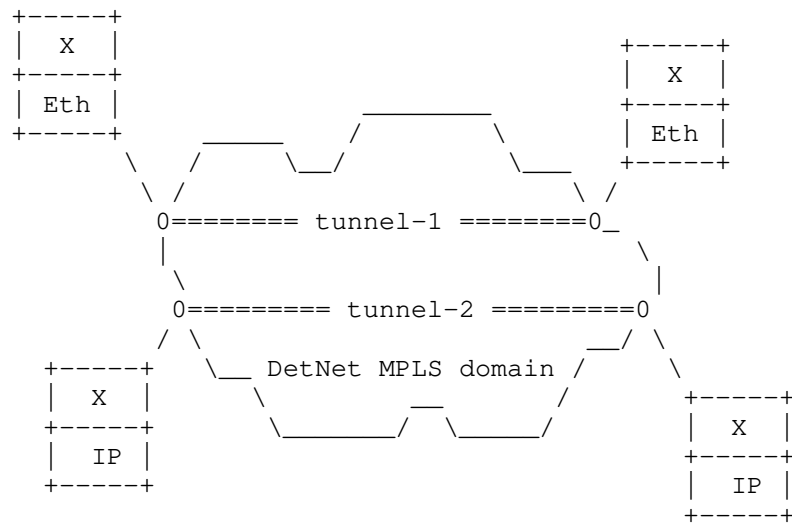


Figure 4: End-Systems and The DetNet MPLS Domain

3.6.5. Sub-Network Considerations

Any of the DetNet service types may be transported by another DetNet service. MPLS nodes may interconnected by different sub-network technologies, which may include point-to-point links. Each of these sub-network technologies need to provide appropriate service to DetNet flows. In some cases, e.g., on dedicated point-to-point links or TDM technologies, all that is required is for a DetNet node to appropriately queue its output traffic. In other cases, DetNet nodes will need to map DetNet flows to the flow semantics (i.e., identifiers) and mechanisms used by an underlying sub-network technology. Figure 5 shows several examples of header formats that can be used to carry DetNet MPLS flows over different sub-network technologies. L2 represent a generic layer-2 encapsulation that might be used on a point-to-point link. TSN represents the encapsulation used on an IEEE 802.1 TSN network, as described in [I-D.ietf-detnet-mpls-over-tsn]. UDP/IP represents the encapsulation used on a DetNet IP PSN, as described in [I-D.ietf-detnet-mpls-over-udp-ip].

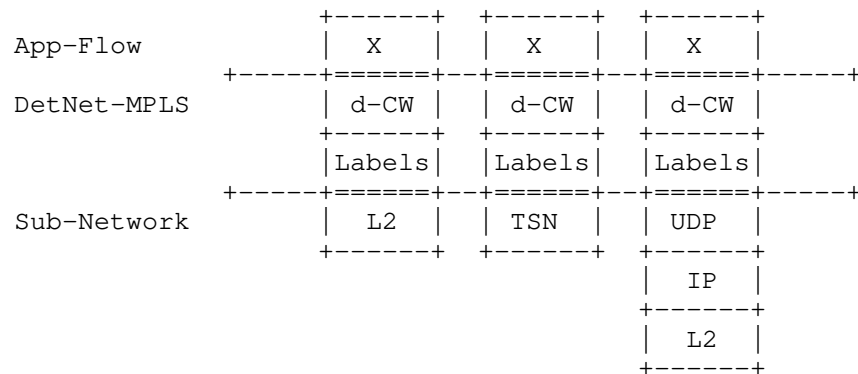


Figure 5: Example DetNet MPLS Sub-Network Formats

4. Controller Plane (Management and Control) Considerations

4.1. DetNet Controller Plane Requirements

While the definition of controller plane for DetNet is out of the scope of this document, there are particular considerations and requirements for such that result from the unique characteristics of the DetNet architecture [I-D.ietf-detnet-architecture] and data plane as defined herein.

The primary requirements of the DetNet controller plane are that it must be able to:

- o Instantiate DetNet flows in a DetNet domain (which may include some or all of explicit path determination, link bandwidth reservations, restricting flows to IEEE 802.1 TSN links, node buffer and other resource reservations, specification of required queuing disciplines along the path, ability to manage bidirectional flows, etc.) as needed for a flow.
- o In the case of MPLS, manage DetNet S-Label and F-Label allocation and distribution, where the DetNet MPLS encapsulation is in use see [I-D.ietf-detnet-mpls].
- o Support DetNet flow aggregation.
- o Advertise static and dynamic node and link resources such as capabilities and adjacencies to other network nodes (for dynamic signaling approaches) or to network controllers (for centralized approaches).

- o Scale to handle the number of DetNet flows expected in a domain (which may require per-flow signaling or provisioning).
- o Provision flow identification information at each of the nodes along the path. Flow identification may differ depending on the location in the network and the DetNet functionality (e.g. transit node vs. relay node).

These requirements, as stated earlier, could be satisfied using distributed control protocol signaling (such as RSVP-TE), centralized network management provisioning mechanisms (such as BGP, PCEP, YANG [I-D.ietf-detnet-flow-information-model], etc.) or hybrid combinations of the two, and could also make use of MPLS-based segment routing.

In the abstract, the results of either distributed signaling or centralized provisioning are equivalent from a DetNet data plane perspective – flows are instantiated, explicit routes are determined, resources are reserved, and packets are forwarded through the domain using the DetNet data plane.

However, from a practical and implementation standpoint, they are not equivalent at all. Some approaches are more scalable than others in terms of signaling load on the network. Some can take advantage of global tracking of resources in the DetNet domain for better overall network resource optimization. Some are more resilient than others if link, node, or management equipment failures occur. While a detailed analysis of the control plane alternatives is out of the scope of this document, the requirements from this document can be used as the basis of a later analysis of the alternatives.

4.2. Generic Controller Plane Considerations

This section covers control plane considerations that are independent of the data plane technology used for DetNet service delivery.

While management plane and control planes are traditionally considered separately, from the Data Plane perspective there is no practical difference based on the origin of flow provisioning information, and the DetNet architecture [I-D.ietf-detnet-architecture] refers to these collectively as the 'Controller Plane'. This document therefore does not distinguish between information provided by distributed control plane protocols, e.g., RSVP-TE [RFC3209] and [RFC3473], or by centralized network management mechanisms, e.g., RestConf [RFC8040], YANG [RFC7950], and the Path Computation Element Communication Protocol (PCEP) [I-D.ietf-pce-pcep-extension-for-pce-controller] or any combination

thereof. Specific considerations and requirements for the DetNet Controller Plane are discussed in Section 4.1.

Each respective data plane document also covers the control plane considerations for that technology. For example [I-D.ietf-detnet-ip] covers IP control plane normative considerations and [I-D.ietf-detnet-mpls] covers MPLS control plane normative considerations.

4.2.1. Flow Aggregation Control

Flow aggregation means that multiple App-flows are served by a single new DetNet flow. There are many techniques to achieve aggregation, for example in case of IP, it can be grouping of IP flows that share 6-tuple attributes or flow identifiers at the DetNet sub-layer. Another example includes aggregation accomplished through the use of hierarchical LSPs in MPLS and tunnels.

Control of aggregation involves a set of procedures listed here. Aggregation may use some or all of these capabilities and the order may vary:

- o Traffic engineering resource collection and distribution:

- Available resources are tracked through control plane or management plane databases and distributed amongst controllers or nodes that can manage resources.

- o Path computation and resource allocation:

- When DetNet services are provisioned or requested one or more paths meeting the requirements are selected and the resources verified and recorded.

- o Resource assignment and data plane co-ordination:

- The assignment of resources along the path depends on the technology and it includes assignment of specific links and coordination of the queuing and other traffic management capabilities such as policing and shaping.

- o Assigned Resource recording and updating:

- Depending on the specific technology, the assigned resources are updated and distributed in the databases, preventing over-subscription.

4.2.2. Explicit Routes

Explicit routes are used to ensure that packets are routed through the resources that have been reserved for them, and hence provide the DetNet application with the required service. A requirement for the DetNet Controller Plane will be the ability to assign a particular identified DetNet IP flow to a path through the DetNet domain that has been assigned the required nodal resources. This provides the appropriate traffic treatment for the flow and also includes particular links as a part of the path that are able to support the DetNet flow. For example, by using IEEE 802.1 TSN links (as discussed in [I-D.ietf-detnet-mpls-over-tsn]) DetNet parameters can be maintained. Further considerations and requirements for the DetNet Controller Plane are discussed in Section 4.1.

Whether configuring, calculating and instantiating these routes is a single-stage or multi-stage process, or in a centralized or distributed manner, is out of scope of this document.

There are several approaches that could be used to provide explicit routes and resource allocation in the DetNet forwarding sub-layer. For example:

- o The path could be explicitly set up by a controller which calculates the path and explicitly configures each node along that path with the appropriate forwarding and resource allocation information.
- o The path could use a distributed control plane such as RSVP [RFC2205] or RSVP-TE [RFC3473] extended to support DetNet IP flows.
- o The path could be implemented using IPv6-based segment routing when extended to support resource allocation.

See Section 4.1 for further discussion of these alternatives. In addition, [RFC2386] contains useful background information on QoS-based routing, and [RFC5575] discusses a specific mechanism used by BGP for traffic flow specification and policy-based routing.

4.2.3. Contention Loss and Jitter Reduction

As discussed in Section 1, this document does not specify the mechanisms needed to eliminate packet contention, packet loss or reduce jitter for DetNet flows at the DetNet forwarding sub-layer. The ability to manage node and link resources to be able to provide these functions is a necessary part of the DetNet controller plane. It is also necessary to be able to control the required queuing

mechanisms used to provide these functions along a flow's path through the network. See [I-D.ietf-detnet-ip] and Section 4.1 for further discussion of these requirements.

4.2.4. Bidirectional Traffic

DetNet applications typically generate bidirectional traffic. IP and MPLS typically treat each direction separately and do not force interdependence of each direction. MPLS has considered bidirectional traffic requirements and the MPLS definitions from [RFC5654] are useful to illustrate terms such as associated bidirectional flows and co-routed bidirectional flows. MPLS defines a point-to-point associated bidirectional LSP as consisting of two unidirectional point-to-point LSPs, one from A to B and the other from B to A, which are regarded as providing a single logical bidirectional forwarding path. This is analogous to standard IP routing. MPLS defines a point-to-point co-routed bidirectional LSP as an associated bidirectional LSP which satisfies the additional constraint that its two unidirectional component LSPs follow the same path (in terms of both nodes and links) in both directions. An important property of co-routed bidirectional LSPs is that their unidirectional component LSPs share fate. In both types of bidirectional LSPs, resource reservations may differ in each direction. The concepts of associated bidirectional flows and co-routed bidirectional flows can also be applied to DetNet IP flows.

While the DetNet IP data plane must support bidirectional DetNet flows, there are no special bidirectional features with respect to the data plane other than the need for the two directions of a co-routed bidirectional flow to take the same path. That is to say that bidirectional DetNet flows are solely represented at the management and control plane levels, without specific support or knowledge within the DetNet data plane. Fate sharing and associated or co-routed bidirectional flows, can be managed at the control level.

DetNet's use of PREOF may increase the complexity of using co-routing bidirectional flows, since if PREOF is used, then the replication points in one direction would have to match the elimination points in the other direction, and vice versa. In such cases the optimal points for these functions in one direction may not match the optimal points in the other, due to network and traffic constraints. Furthermore, due to the per packet service protection nature, bidirectional forwarding per packet may not be ensured. The first packet of received member flows is selected by the elimination function independently of which path it has taken through the network.

Control and management mechanisms need to support bidirectional flows, but the specification of such mechanisms are out of scope of this document. An example control plane solution for MPLS can be found in [RFC3473], [RFC6387] and [RFC7551]. These requirements are included in Section 4.1.

4.3. Packet Replication, Elimination, and Ordering (PREOF)

The controller plane protocol solution required for managing the PREOF processing is outside the scope of this document. That said, it should be noted that the ability to determine, for a particular flow, optimal packet replication and elimination points in the DetNet domain requires explicit support. There may be capabilities that can be used, or extended, for example GMPLS end-to-end recovery [RFC4872] and GMPLS segment recovery [RFC4873].

5. Security Considerations

Security considerations for DetNet are described in detail in [I-D.ietf-detnet-security]. General security considerations are described in [I-D.ietf-detnet-architecture]. This section considers general security considerations applicable to all data planes.

Security aspects which are unique to DetNet are those whose aim is to provide the specific quality of service aspects of DetNet, which are primarily to deliver data flows with extremely low packet loss rates and bounded end-to-end delivery latency.

The primary considerations for the data plane is to maintain integrity of data and delivery of the associated DetNet service traversing the DetNet network. Application flows can be protected through whatever means is provided by the underlying technology. For example, encryption may be used, such as that provided by IPSec [RFC4301] for IP flows and/or by an underlying sub-net using MACSec [IEEE802.1AE-2018] for Ethernet (Layer-2) flows.

At the management and control level DetNet flows are identified on a per-flow basis, which may provide controller plane attackers with additional information about the data flows (when compared to controller planes that do not include per-flow identification). This is an inherent property of DetNet which has security implications that should be considered when determining if DetNet is a suitable technology for any given use case.

To provide uninterrupted availability of the DetNet service, provisions can be made against DOS attacks and delay attacks. To protect against DOS attacks, excess traffic due to malicious or malfunctioning devices can be prevented or mitigated, for example

through the use of existing mechanism such as policing and shaping applied at the input of a DetNet domain. To prevent DetNet packets from being delayed by an entity external to a DetNet domain, DetNet technology definition can allow for the mitigation of Man-In-The-Middle attacks, for example through use of authentication and authorization of devices within the DetNet domain.

In order to prevent or mitigate DetNet attacks on other networks via flow escape, edge devices can for example use existing mechanism such as policing and shaping applied at the output of a DetNet domain.

6. IANA Considerations

This document makes no IANA requests.

7. Acknowledgements

The authors wish to thank Pat Thaler, Norman Finn, Loa Anderson, David Black, Rodney Cummings, Ethan Grossman, Tal Mizrahi, David Mozes, Craig Gunther, George Swallow, Yuanlong Jiang and Carlos J. Bernardos for their various contributions to this work.

8. References

8.1. Normative References

- [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001, <<https://www.rfc-editor.org/info/rfc3209>>.
- [RFC3473] Berger, L., Ed., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions", RFC 3473, DOI 10.17487/RFC3473, January 2003, <<https://www.rfc-editor.org/info/rfc3473>>.
- [RFC4385] Bryant, S., Swallow, G., Martini, L., and D. McPherson, "Pseudowire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN", RFC 4385, DOI 10.17487/RFC4385, February 2006, <<https://www.rfc-editor.org/info/rfc4385>>.

8.2. Informative References

- [I-D.ietf-detnet-architecture] Finn, N., Thubert, P., Varga, B., and J. Farkas, "Deterministic Networking Architecture", draft-ietf-detnet-architecture-13 (work in progress), May 2019.

- [I-D.ietf-detnet-flow-information-model]
Farkas, J., Varga, B., Cummings, R., Jiang, Y., and D. Fedyk, "DetNet Flow Information Model", draft-ietf-detnet-flow-information-model-05 (work in progress), September 2019.
- [I-D.ietf-detnet-ip]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: IP", draft-ietf-detnet-ip-01 (work in progress), July 2019.
- [I-D.ietf-detnet-mpls]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: MPLS", draft-ietf-detnet-mpls-01 (work in progress), July 2019.
- [I-D.ietf-detnet-mpls-over-tsn]
Varga, B., Farkas, J., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: MPLS over IEEE 802.1 Time Sensitive Networking (TSN)", draft-ietf-detnet-mpls-over-tsn-00 (work in progress), May 2019.
- [I-D.ietf-detnet-mpls-over-udp-ip]
Varga, B., Farkas, J., Berger, L., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: MPLS over UDP/IP", draft-ietf-detnet-mpls-over-udp-ip-02 (work in progress), October 2019.
- [I-D.ietf-detnet-security]
Mizrahi, T., Grossman, E., Hacker, A., Das, S., Dowdell, J., Austad, H., Stanton, K., and N. Finn, "Deterministic Networking (DetNet) Security Considerations", draft-ietf-detnet-security-05 (work in progress), August 2019.
- [I-D.ietf-pce-pcep-extension-for-pce-controller]
Zhao, Q., Li, Z., Negi, M., and C. Zhou, "PCEP Procedures and Protocol Extensions for Using PCE as a Central Controller (PCECC) of LSPs", draft-ietf-pce-pcep-extension-for-pce-controller-02 (work in progress), July 2019.
- [I-D.ietf-spring-srv6-network-programming]
Filsfils, C., Camarillo, P., Leddy, J., daniel.voyer@bell.ca, d., Matsushima, S., and Z. Li, "SRv6 Network Programming", draft-ietf-spring-srv6-network-programming-05 (work in progress), October 2019.

- [IEEE802.1AE-2018]
IEEE Standards Association, "IEEE Std 802.1AE-2018 MAC Security (MACsec)", 2018,
<<https://ieeexplore.ieee.org/document/8585421>>.
- [IEEE802.1TSNTG]
IEEE Standards Association, "IEEE 802.1 Time-Sensitive Networking Task Group", <<http://www.ieee802.org/1/tsn>>.
- [RFC2205] Braden, R., Ed., Zhang, L., Berson, S., Herzog, S., and S. Jamin, "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", RFC 2205, DOI 10.17487/RFC2205, September 1997, <<https://www.rfc-editor.org/info/rfc2205>>.
- [RFC2386] Crawley, E., Nair, R., Rajagopalan, B., and H. Sandick, "A Framework for QoS-based Routing in the Internet", RFC 2386, DOI 10.17487/RFC2386, August 1998, <<https://www.rfc-editor.org/info/rfc2386>>.
- [RFC3670] Moore, B., Durham, D., Strassner, J., Westerinen, A., and W. Weiss, "Information Model for Describing Network Device QoS Datapath Mechanisms", RFC 3670, DOI 10.17487/RFC3670, January 2004, <<https://www.rfc-editor.org/info/rfc3670>>.
- [RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <<https://www.rfc-editor.org/info/rfc4301>>.
- [RFC4872] Lang, J., Ed., Rekhter, Y., Ed., and D. Papadimitriou, Ed., "RSVP-TE Extensions in Support of End-to-End Generalized Multi-Protocol Label Switching (GMPLS) Recovery", RFC 4872, DOI 10.17487/RFC4872, May 2007, <<https://www.rfc-editor.org/info/rfc4872>>.
- [RFC4873] Berger, L., Bryskin, I., Papadimitriou, D., and A. Farrel, "GMPLS Segment Recovery", RFC 4873, DOI 10.17487/RFC4873, May 2007, <<https://www.rfc-editor.org/info/rfc4873>>.
- [RFC5575] Marques, P., Sheth, N., Raszuk, R., Greene, B., Mauch, J., and D. McPherson, "Dissemination of Flow Specification Rules", RFC 5575, DOI 10.17487/RFC5575, August 2009, <<https://www.rfc-editor.org/info/rfc5575>>.
- [RFC5654] Niven-Jenkins, B., Ed., Brungard, D., Ed., Betts, M., Ed., Sprecher, N., and S. Ueno, "Requirements of an MPLS Transport Profile", RFC 5654, DOI 10.17487/RFC5654, September 2009, <<https://www.rfc-editor.org/info/rfc5654>>.

- [RFC6387] Takacs, A., Berger, L., Caviglia, D., Fedyk, D., and J. Meuric, "GMPLS Asymmetric Bandwidth Bidirectional Label Switched Paths (LSPs)", RFC 6387, DOI 10.17487/RFC6387, September 2011, <<https://www.rfc-editor.org/info/rfc6387>>.
- [RFC7551] Zhang, F., Ed., Jing, R., and R. Gandhi, Ed., "RSVP-TE Extensions for Associated Bidirectional Label Switched Paths (LSPs)", RFC 7551, DOI 10.17487/RFC7551, May 2015, <<https://www.rfc-editor.org/info/rfc7551>>.
- [RFC7657] Black, D., Ed. and P. Jones, "Differentiated Services (Diffserv) and Real-Time Communication", RFC 7657, DOI 10.17487/RFC7657, November 2015, <<https://www.rfc-editor.org/info/rfc7657>>.
- [RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <<https://www.rfc-editor.org/info/rfc7950>>.
- [RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <<https://www.rfc-editor.org/info/rfc8040>>.
- [RFC8227] Cheng, W., Wang, L., Li, H., van Helvoort, H., and J. Dong, "MPLS-TP Shared-Ring Protection (MSRP) Mechanism for Ring Topology", RFC 8227, DOI 10.17487/RFC8227, August 2017, <<https://www.rfc-editor.org/info/rfc8227>>.

Authors' Addresses

Balazs Varga (editor)
Ericsson
Magyar Tudosok krt. 11.
Budapest 1117
Hungary

Email: balazs.a.varga@ericsson.com

Janos Farkas
Ericsson
Magyar Tudosok krt. 11.
Budapest 1117
Hungary

Email: janos.farkas@ericsson.com

Lou Berger
LabN Consulting, L.L.C.

Email: lberger@labn.net

Don Fedyk
LabN Consulting, L.L.C.

Email: dfedyk@labn.net

Andrew G. Malis
Independent

Email: agmalis@gmail.com

Stewart Bryant
Futurewei Technologies

Email: stewart.bryant@gmail.com

Jouni Korhonen

Email: jouni.nospam@gmail.com

DetNet
Internet-Draft
Intended status: Standards Track
Expires: April 29, 2020

J. Farkas
B. Varga
Ericsson
R. Cummings
National Instruments
Y. Jiang
Huawei Technologies Co., Ltd.
D. Fedyk
LabN Consulting, L.L.C.
October 27, 2019

DetNet Flow Information Model
draft-ietf-detnet-flow-information-model-06

Abstract

This document describes flow and service information model for Deterministic Networking (DetNet). These models are defined for IP and MPLS DetNet data planes

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 April 29, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect

to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
1.1. Goals	5
1.2. Non Goals	6
2. Terminology	6
2.1. Terms Used in This Document	6
2.2. Abbreviations	7
2.3. Naming Conventions	7
2.4. Requirements Language	7
3. DetNet Domain and its Modeling	7
3.1. DetNet Service Overview	7
3.2. Reference Points Used in Modeling	8
3.3. Information Elements	8
4. App-flow Related Parameters	9
4.1. App-flow Characteristics	9
4.2. App-flow Requirements	9
5. DetNet Flow Related Parameters	10
5.1. Management ID of the DetNet Flow	10
5.2. Payload type of the DetNet Flow	11
5.3. Format of the DetNet Flow	11
5.4. Identification and Specification of DetNet Flows	11
5.4.1. DetNet MPLS Flow Identification and Specification	11
5.4.2. DetNet IP Flow Identification and Specification	11
5.5. Traffic Specification of the DetNet Flow	11
5.6. Endpoints of the DetNet Flow	12
5.7. Rank of the DetNet Flow	12
5.8. Status of the DetNet Flow	13
5.9. Requirements of the DetNet Flow	13
5.9.1. Minimum Bandwidth of the DetNet Flow	14
5.9.2. Maximum Latency of the DetNet Flow	14
5.9.3. Maximum Latency Variation of the DetNet Flow	14
5.9.4. Maximum Loss of the DetNet Flow	14
5.9.5. Maximum Consecutive Loss of the DetNet Flow	14
5.9.6. Maximum Misordering Tolerance of the DetNet Flow	14
5.10. BiDir requirement of the DetNet Flow	14
6. DetNet Service Related Parameters	15
6.1. Management ID of the DetNet service	15
6.2. Delivery Type of the DetNet service	15
6.3. Delivery Profile of the DetNet Service	15
6.3.1. Minimum Bandwidth of the DetNet Service	15
6.3.2. Maximum Latency of the DetNet Service	16
6.3.3. Maximum Latency Variation of the DetNet Service	16

6.3.4. Maximum Loss of the DetNet Service	16
6.3.5. Maximum Consecutive Loss of the DetNet Service . . .	16
6.3.6. Maximum Misordering Tolerance of the DetNet Service .	16
6.4. Connectivity Type of the DetNet Service	16
6.5. BiDir requirement of the DetNet Service	16
6.6. Rank of the DetNet Service	17
6.7. Status of the DetNet Service	17
7. Flow Specific Operations	18
7.1. Join Operation	18
7.2. Leave Operation	18
7.3. Modify Operation	18
8. Summary	19
9. IANA Considerations	19
10. Security Considerations	19
11. References	19
11.1. Normative References	19
11.2. Informative References	20
Authors' Addresses	21

1. Introduction

Deterministic Networking (DetNet) provides a capability to carry specified unicast or multicast data flows for real-time applications with extremely low packet loss rates and assured maximum end-to-end delivery latency. A description of the general background and concepts of DetNet can be found in [I-D.ietf-detnet-architecture].

This document describes the Detnet Flow Service Information Model. For reference [RFC3444] describes the rational behind Information Models in general. This document describes the Flow and Service information models for operators and users to understand Detnet services, and for implementors as a guide to the functionality required by Detnet services.

The DetNet Architecture treats the DetNet related data plane functions decomposed into two sub-layers: a service sub-layer and a forwarding sub-layer. The service sub-layer is used to provide DetNet service protection and reordering. The forwarding sub-layer provides resource allocation (to ensure low loss, assured latency, and limited out-of-order delivery) and leverages Traffic Engineering mechanisms.

In the IETF DetNet service utilizes IP or MPLS and DetNet is currently defined for IP and MPLS networks as shown in Figure 1 based on Figure 2 and Figure 3 of [I-D.ietf-detnet-data-plane-framework]. IEEE 802.1 Time Sensitive Networking (TSN) utilizes Ethernet and is defined over Ethernet networks. A DetNet flow includes one or more App-flow(s) as payload. App-flows can be Ethernet, MPLS, or IP

flows, which impacts which header fields are utilized to identify a flow. DetNet flows are identified by the DetNet encapsulation of App-flow(s) (e.g., MPLS labels, IP 6-tuple etc.). In some scenarios App-flow and DetNet flow look similar on the wire (e.g., L3 App-flow over a DetNet IP network).

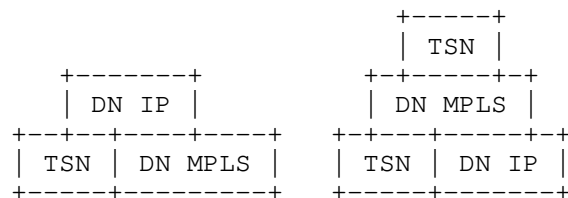


Figure 1: DetNet Service Examples as per Data Plane Framework

As shown in Figure 1 as per [I-D.ietf-detnet-data-plane-framework] a DetNet flow can be treated as an application level flow (App-flow) e.g., at DetNet flow aggregation or in a sub-network that interconnects DetNet nodes.

The DetNet flow and service information model provided by this document contains both DetNet flow and App-flow specific information in an integrated fashion.

In a given network scenario three information models can be distinguished:

- o Flow models that describe characteristics of data flows. These models describe in detail all relevant aspects of a flow that are needed to support the flow properly by the network between the source and the destination(s).
- o Service models that describe characteristics of services being provided for data flows over a network. These models can be treated as a network operator independent information model.
- o Configuration models that describe in detail the settings required on network nodes to provide a data flow proper service.

Service and flow information models are used between the user and the network operator. Configuration information models are used between the management/control plane entity of the network and the network nodes. They are shown in Figure 2.

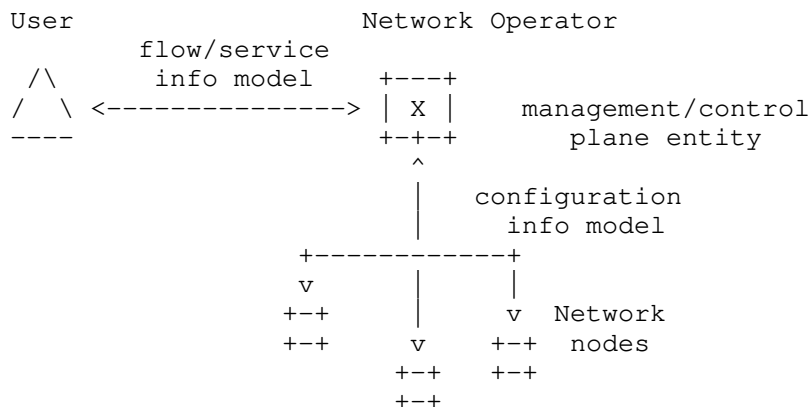


Figure 2: Usage of Information models (flow, service and configuration)

DetNet flow and service information model is based on [I-D.ietf-detnet-architecture] and on the concept of data model specified by [IEEE8021Qcc]. Furthermore, the origination of the DetNet flow information model was the flow identification possibilities described in [IEEE8021CB], which is used by [IEEE8021Qcc] as well. In addition to the TSN data model, [IEEE8021Qcc] also specifies configuration of TSN features (e.g., traffic scheduling specified by [IEEE8021Qbv]). The common architecture and flow model, allow configured features to be consistent in certain deployment scenarios, e.g., when the network that provides the DetNet service includes both L3 and L2 network segments.

1.1. Goals

As expressed in the [IETFDetNet] Charter, the DetNet WG collaborates with IEEE 802.1 TSN in order to define a common architecture for both Layer 2 and Layer 3. This is beneficial for several reasons, e.g., in order to simplify implementations and maintain consistency across diverse networks. The flow and service information models are also aligned for those reasons. Therefore, the DetNet flow and service information models described in this document are based on [IEEE8021Qcc], which is an amendment to [IEEE8021Q].

This document specifies flow and service information models only.

1.2. Non Goals

This document (this revision) does not specify flow data models or DetNet configuration. Therefore, the goals of this document differ from the goals of [IEEE8021Qcc], which also specifies the TSN data model and configuration of certain TSN features.

2. Terminology

2.1. Terms Used in This Document

This document uses the terminology established in the DetNet architecture [I-D.ietf-detnet-architecture] and the DetNet Data Plane Framework [I-D.ietf-detnet-data-plane-framework]. The reader is assumed to be familiar with these documents and any terminology defined therein. The DetNet <=> TSN dictionary of [I-D.ietf-detnet-architecture] is used to perform translation from [IEEE8021Qcc] to this document.

The following terminology is used in accordance with [I-D.ietf-detnet-architecture]:

App-flow	The payload (data) carried over a DetNet service.
DetNet flow	A DetNet flow is a sequence of packets which conform uniquely to a flow identifier, and to which the DetNet service is to be provided. It includes any DetNet headers added to support the DetNet service and forwarding sub-layers.

The following terminology is introduced in this document:

Source	Reference point for an App-flow, where the flow starts.
Destination	Reference point for an App-flow, where the flow terminates.
DN Ingress	Reference point for the start of a DetNet flow. Networking technology specific encapsulation may be added here to the served App-flow(s).
DN Egress	Reference point for the termination of a DetNet flow. Networking technology specific encapsulation may be removed here from the served App-flow(s).

2.2. Abbreviations

The following abbreviations are used in this document:

DetNet	Deterministic Networking.
DN	DetNet.
MPLS	Multiprotocol Label Switching.
PSN	Packet Switched Network.
TSN	Time-Sensitive Networking.

2.3. Naming Conventions

The following naming conventions were used for naming information model components in this document. It is recommended that extensions of the model use the same conventions.

- o Names SHOULD be descriptive.
- o Names MUST start with uppercase letters.
- o Composed names MUST use capital letters for the first letter of each component. All other letters are lowercase, even for acronyms. Exceptions are made for acronyms containing a mixture of lowercase and capital letters, such as IPv6. Example composed names are SourceMacAddress and DestinationIPv6Address.

2.4. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. DetNet Domain and its Modeling

3.1. DetNet Service Overview

The DetNet service can be defined as a service that provides a capability to carry a unicast or a multicast data flow for an application with constrained requirements on network performance, e.g., low packet loss rate and/or latency.

Figure 5 and Figure 8 in [I-D.ietf-detnet-architecture] show the DetNet service related reference points and main components.

3.2. Reference Points Used in Modeling

From service design perspective a fundamental question is the location of the service/flow endpoints, i.e., where the service/flow starts and ends.

App-flow specific reference points are the Source (where it starts) and the Destination (where it terminates). Similarly a DetNet flow has reference points termed DN Ingress (where a DetNet flow starts) and DN Egress (where a DetNet flow ends). These reference points may coexist in the same node (e.g., in a DetNet IP end system). DN Ingress and DN Egress reference points are intermediate reference points for a served App-flow.

All reference points are assumed in this document to be packet-based reference points. A DN Ingress may add and a DN Egress may remove networking technology specific encapsulation to/from the served App-flow(s) (e.g., MPLS label(s), UDP and IP headers).

3.3. Information Elements

The DetNet flow information model and the service model relies on three groups of information elements:

- o App-flow related parameters: these describe the App-flow characteristics (e.g., identification, encapsulation, traffic specification, endpoints, status, etc.) and the App-flow service expectations (e.g., delay, loss, etc.).
- o DetNet flow related parameters: these describe the DetNet flow characteristics (e.g., identification, format, traffic specification, endpoints, rank, etc.).
- o DetNet service related parameters: these describe the expected service characteristics (e.g., delivery type, connectivity delay/loss, status, rank, etc.).

In the information model a DetNet flow contains one or more (aggregated) App-flows (N:1 mapping). During DetNet aggregation the aggregated DetNet flows are treated simply as App-flows and the aggregate is the DetNet flow, which provides N:1 mapping. Similarly, there is an aggregated many to one relationship for the DetNet flow(s) to the DetNet Service.

4. App-flow Related Parameters

When Deterministic service is required by time/loss sensitive application(s) running on an end system during communication with its peer(s), the resulting data exchange has various requirements on delay and/or loss parameters.

4.1. App-flow Characteristics

App-flow characteristics are described by the following parameters:

- o FlowID: a unique (management) identifier of the App-flow. It can be used to define the N:1 mapping of App-flows to a DetNet flow.
- o FlowType: set by the encapsulation format of the flow. It can be Ethernet (TSN), MPLS, or IP.
- o DataFlowSpecification: a flow descriptor, defining which packets belongs to a flow using, specific packet header fields such as src-addr, dst-addr, label, VLAN-ID, etc.
- o TrafficSpecification: a flow descriptor, defining traffic parameters such as packet size, transmission time interval, and maximum packets per time interval.
- o FlowEndpoints: delineate the start and termination reference points of the App-flow by pointing to the source interface/node and destination interface(s)/node(s).
- o FlowStatus: indicates the status of the App-flow with respect to the establishment of the flow by the connected network, e.g., ready, failed, etc.
- o FlowRank: indicates the rank of this flow relative to other flows in the connected network.

4.2. App-flow Requirements

App-flow requirements are described by the following parameters:

- o FlowRequirements: defines the attributes of the App-flow regarding bandwidth, latency, latency variation, loss, and misordering tolerance.
- o FlowBiDir: defines the data path requirement of the App-flow whether it must share the same data path and physical path for both directions through the network, e.g., to provide congruent paths in the two directions.

5. DetNet Flow Related Parameters

The Data model specified by [IEEE8021Qcc] describes data flows using TSN service as periodic flows with fix packet size (i.e., Constant Bit Rate (CBR) flows) or with variable packet size. The same concept is applied for flows using DetNet service.

Latency and loss parameters are correlated because the effect of late delivery can result data loss for an application. However, not all applications require hard limits on both latency and loss. For example, some real-time applications allow graceful degradation if loss happens (e.g., sample-based data processing, media distribution). Some other applications may require high-bandwidth connections that make use of packet replication techniques which are economically challenging or even impossible. Some applications may not tolerate loss, but are not delay sensitive (e.g., bufferless sensors). Time or loss sensitive applications may have somewhat special requirements especially for loss (e.g., no loss over two consecutive communication cycles; very low outage time, etc.).

DetNet flows have the following attributes:

- a. DnFlowID (Section 5.1)
- b. DnPayloadType (Section 5.2)
- c. DnFlowFormat (Section 5.3)
- d. DnFlowSpecification (Section 5.4)
- e. DnTrafficSpecification (Section 5.5)
- f. DnFlowEndpoints (Section 5.6)
- g. DnFlowRank (Section 5.7)
- h. DnFlowStatus (Section 5.8)

DetNet flows have the following requirement attributes:

- o DnFlowRequirements (Section 5.9)
- o DnFlowBiDir (Section 5.10)

Flow attributes are described in the following sections.

5.1. Management ID of the DetNet Flow

A unique (management) identifier is needed for each DetNet flow within the DetNet domain. It is specified by DnFlowID. It can be used to define the many to one mapping of DetNet flows to a DetNet service.

5.2. Payload type of the DetNet Flow

The DnPayloadType attribute is set according to the encapsulated App-flow format. The attribute can be Ethernet, MPLS, or IP.

5.3. Format of the DetNet Flow

The DnFlowFormat attribute is set according to the DetNet PSN technology. The attribute can be MPLS or IP.

5.4. Identification and Specification of DetNet Flows

Identification options for DetNet flows at the Ingress/Egress and within the DetNet domain are specified as follows; see Section 5.4.1 for DetNet MPLS flows and Section 5.4.2 for DetNetw IP flows.

5.4.1. DetNet MPLS Flow Identification and Specification

The identification of DetNet MPLS flows within the DetNet domain is based on the MPLS context in the service information model. The attributes are specific to the MPLS forwarding paradigm within the DetNet domain [I-D.ietf-detnet-mpls]. DetNetwork MPLS flows can be identified and specified by the following attributes:

- a. SLabel
- b. FLabelStack

5.4.2. DetNet IP Flow Identification and Specification

DetNet IP flows can be identified and specified by the following attributes (6-tuple) [I-D.ietf-detnet-ip]:

- a. SourceIpAddress
- b. DestinationIpAddress
- c. IPv6FlowLabel
- d. Dscp (attribute)
- e. Protocol
- f. SourcePort
- g. DestinationPort
- h. IPSecSpi

5.5. Traffic Specification of the DetNet Flow

DnTrafficSpecification attributes specify how the DN Ingress transmits packets for the DetNet flow. This is effectively the promise/request of the DN Ingress to the network. The network uses this traffic specification to allocate resources and adjust queue parameters in network nodes.

TrafficSpecification has the following attributes:

- a. Interval: the period of time in which the traffic specification is specified.
- b. MaxPacketsPerInterval: the maximum number of packets that the Ingress will transmit in one Interval.
- c. MaxPayloadSize: the maximum payload size that the Ingress will transmit.

These attributes can be used to describe any type of traffic (e.g., CBR, VBR, etc.) and can be used during resource allocation to represent worst case scenarios.

[[Editor's note (to be removed from a future revision): Further optional attributes can be considered to achieve more efficient resource allocation. Such optional attributes might be worth for flows with soft requirements (i.e., the flow is only loss sensitive or only delay sensitive, but not both delay-and-loss sensitive). Possible options how to extend DnTrafficSpecification attributes is for further discussion.]]

5.6. Endpoints of the DetNet Flow

The DnFlowEndpoints attribute defines the starting and termination reference points of the DetNet flow by pointing to the ingress interface/node and egress interface(s)/node(s). Depending on the network scenario it defines an interface or a node. Interface can be defined for example if the App-flow is a TSN Stream and it is received over a well defined UNI interface. For example, for App-flows with MPLS encapsulation defining an ingress node is more common when per platform label space is used.

5.7. Rank of the DetNet Flow

The DnFlowRank attribute provides the rank of this flow relative to other flows in the DetNet domain. Rank (range: 0-255) is used by the DetNet domain to decide which flows can and cannot exist when network resources reach their limit. Rank is used to help to determine which flows can be bumped (i.e., removed from node configuration thereby releasing its resources) if for example a port of a node becomes oversubscribed (e.g., due to network re-configuration).

5.8. Status of the DetNet Flow

DnFlowStatus provides the status of the DetNet flow with respect to the establishment of the flow by the DetNet domain.

The DnFlowStatus SHALL include the following attributes:

- a. DnIngressStatus is an enumeration for the status of the flow's Ingress reference point:
 - * None: no Ingress.
 - * Ready: Ingress is ready.
 - * Failed: Ingress failed.
 - * OutOfService: Administratively blocked.
- b. DnEgressStatus is an enumeration for the status of the flow's Egress reference points:
 - * None: no Egress.
 - * Ready: all Egresses are ready.
 - * PartialFailed: One or more Egress ready, and one or more Egress failed. The DetNet flow can be used if the Ingress is Ready.
 - * Failed: All Egresses failed.
 - * OutOfService: Administratively blocked.
- c. FailureCode: A non-zero code that specifies the error if the DetNet flow encounters a failure (e.g., packet replication and elimination is requested but not possible, or DnIngressStatus is Failed, or DnEgressStatus is Failed, or DnEgressStatus is PartialFailed).

[[Editor's note (to be removed from a future revision): FailureCodes to be defined for DetNet. Table 46-1 of [IEEE8021Qcc] describes TSN failure codes.]]

5.9. Requirements of the DetNet Flow

DnFlowRequirements specifies requirements to ensure the service level desired for the DetNet flow.

The DnFlowRequirements includes the following attributes:

- a. MinBandwidth(Section 5.9.1)
- b. MaxLatency(Section 5.9.2)
- c. MaxLatencyVariation(Section 5.9.3)
- d. MaxLoss(Section 5.9.4)
- e. MaxConsecutiveLossTolerance(Section 5.9.5)

f. MaxMisordering(Section 5.9.6)

5.9.1. Minimum Bandwidth of the DetNet Flow

MinBandwidth is the minimum bandwidth that has to be guaranteed for the DetNet flow. MinBandwidth is specified in octets per second.

5.9.2. Maximum Latency of the DetNet Flow

MaxLatency is the maximum latency from Ingress to Egress(es) for a single packet of the DetNet flow. MaxLatency is specified as an integer number of nanoseconds.

5.9.3. Maximum Latency Variation of the DetNet Flow

MaxLatencyVariation is the difference between the minimum and the maximum end-to-end one-way latency. MaxLatencyVariation is specified as an integer number of nanoseconds.

5.9.4. Maximum Loss of the DetNet Flow

MaxLoss defines the maximum Packet Loss Ratio (PLR) requirement for the DetNet flow between the Ingress and Egress(es).

5.9.5. Maximum Consecutive Loss of the DetNet Flow

Some applications have special loss requirement, such as MaxConsecutiveLossTolerance. The maximum consecutive loss tolerance parameter describes the maximum number of consecutive packets whose loss can be tolerated. The maximum consecutive loss tolerance can be measured for example based on sequence number.

5.9.6. Maximum Misordering Tolerance of the DetNet Flow

MaxMisordering describes the tolerable maximum number of packets that can be received out of order. The maximum allowed misordering can be measured for example based on sequence number. The value zero for the maximum allowed misordering indicates that in order delivery is required, misordering cannot be tolerated.

5.10. BiDir requirement of the DetNet Flow

DnFlowBiDir attribute defines the requirement that the flow and the corresponding reverse direction flow must share the same path (links and nodes) through the routed or switch network in the DetNet domain, e.g., to provide congruent paths in the two directions that share fate and path characteristics.

6. DetNet Service Related Parameters

DetNet service have the following attributes:

- a. DnServiceID (Section 6.1)
- b. DnServiceDeliveryType (Section 6.2)
- c. DnServiceDeliveryProfile (Section 6.3)
- d. DnServiceConnectivity (Section 6.4)
- e. DnServiceBiDir (Section 6.5)
- f. DnServiceRank (Section 6.6)
- g. DnServiceStatus (Section 6.7)

Service attributes are described in the following sections.

6.1. Management ID of the DetNet service

A unique (management) identifier for each DetNet service within the DetNet domain. It can be used to define the many to one mapping of DetNet flows to a DetNet service.

6.2. Delivery Type of the DetNet service

The DnServiceDeliveryType attribute is set according to the payload of the served DetNet flow (i.e., the encapsulated App-flow format). The attribute can be Ethernet, MPLS, or IP.

6.3. Delivery Profile of the DetNet Service

DnServiceDeliveryProfile specifies delivery profile to ensure proper serving of the DetNet flow.

The DnServiceDeliveryProfile includes the following attributes:

- a. MinBandwidth(Section 6.3.1)
- b. MaxLatency(Section 6.3.2)
- c. MaxLatencyVariation(Section 6.3.3)
- d. MaxLoss(Section 6.3.4)
- e. MaxConsecutiveLossTolerance(Section 6.3.5)
- f. MaxMisordering(Section 6.3.6)

6.3.1. Minimum Bandwidth of the DetNet Service

MinBandwidth is the minimum bandwidth that has to be guaranteed for the DetNet service. MinBandwidth is specified in octets per second.

6.3.2. Maximum Latency of the DetNet Service

MaxLatency is the maximum latency from Ingress to Egress(es) for a single packet of the DetNet flow. MaxLatency is specified as an integer number of nanoseconds.

6.3.3. Maximum Latency Variation of the DetNet Service

MaxLatencyVariation is the difference between the minimum and the maximum end-to-end one-way latency. MaxLatencyVariation is specified as an integer number of nanoseconds.

6.3.4. Maximum Loss of the DetNet Service

MaxLoss defines the maximum Packet Loss Ratio (PLR) parameter for the DetNet service between the Ingress and Egress(es) of the DetNet domain.

6.3.5. Maximum Consecutive Loss of the DetNet Service

Some applications have special loss requirement, such as MaxConsecutiveLossTolerance. The maximum consecutive loss tolerance parameter describes the maximum number of consecutive packets whose loss can be tolerated. The maximum consecutive loss tolerance can be measured for example based on sequence number.

6.3.6. Maximum Misordering Tolerance of the DetNet Service

MaxMisordering describes the tolerable maximum number of packets that can be received out of order. The maximum allowed misordering can be measured for example based on sequence number. The value zero for the maximum allowed misordering indicates that in order delivery is required, misordering cannot be tolerated.

6.4. Connectivity Type of the DetNet Service

Two connectivity types are distinguished: point-to-point (p2p) and point-to-multipoint (p2mp). Connectivity type p2mp is created by a transport layer function (e.g., p2mp LSP). (Note: mp2mp connectivity is a superposition of p2mp connections.)

6.5. BiDir requirement of the DetNet Service

The DnServiceBiDir attribute defines the requirement that the flow and the corresponding reverse direction flow must share the same path (links and nodes) through the routed or switch network in the DetNet domain, e.g., to provide congruent paths in the two directions that share fate and path characteristics.

6.6. Rank of the DetNet Service

The DnServiceRank attribute provides the rank of a service instance relative to other services in the DetNet domain. DnServiceRank (range: 0-255) is used by the network in case of network resource limitation scenarios.

6.7. Status of the DetNet Service

DnServiceStatus information group includes elements that specify the status of the service specific state of the DetNet domain. This information group informs the user whether or not the service is ready for use.

The DnServiceStatus SHALL include the following attributes:

- a. DnServiceIngressStatus is an enumeration for the status of the service's Ingress:
 - * None: no Ingress.
 - * Ready: Ingress is ready.
 - * Failed: Ingress failed.
 - * OutOfService: Administratively blocked.
- b. DnServiceEgressStatus is an enumeration for the status of the service's Egress:
 - * None: no Egress.
 - * Ready: all Egresses are ready.
 - * PartialFailed: One or more Egress ready, and one or more Egress failed. The DetNet flow can be used if the Ingress is Ready.
 - * Failed: All Egresses failed.
 - * OutOfService: Administratively blocked.
- c. DnServiceFailureCode: A non-zero code that specifies the error if the DetNet service encounters a failure (e.g., packet replication and elimination is requested but not possible, or DnServiceIngressStatus is Failed, or DnServiceEgressStatus is Failed, or DnServiceEgressStatus is PartialFailed).

[[Editor's note (to be removed from a future revision):
DnServiceFailureCodes to be defined for DetNet service. Table 46-1 of [IEEE8021Qcc] describes TSN failure codes.]]

7. Flow Specific Operations

The DetNet flow information model relies on three high level information groups:

- o DnIngress: The DnIngress information group includes elements that specify the source for a single DetNet flow. This information group is applied from the user of the DetNet service to the network.
- o DnEgress: The DnEgress information group includes elements that specify the destination for a single DetNet flow. This information group is applied from the user of the DetNet service to the network.
- o DnFlowStatus: The status information group includes elements that specify the status of the flow in the network. This information group is applied from the network to the user of the DetNet service. This information group informs the user whether or not the DetNet flow is ready for use.

There are three possible operations for each DetNet flow with respect to its DetNet service at a DN Ingress or a DN Egress (similarly to App-flows at a Source or a Destination):

- o Join: DN Ingress/DN Egress intends to join the flow.
- o Leave: DN Ingress/DN Egress intends to leave the flow.
- o Modify: DN Ingress/DN Egress intends to change the flow.

7.1. Join Operation

For the join operation, the DnFlowSpecification, DnFlowRank, DnFlowEndpoint, and DnTrafficSpecification SHALL be included within the DnIngress or DnEgress information group. For the join operation, the DnServiceRequirements groups MAY be included.

7.2. Leave Operation

For the leave operation, the DnFlowSpecification and DnFlowEndpoint SHALL be included within the DnIngress or DnEgress information group.

7.3. Modify Operation

For the modify operation, the DnFlowSpecification, DnFlowRank, DnFlowEndpoint, and DnTrafficSpecification SHALL be included within the DnIngress or DnEgress information group. For the join operation, the DnServiceRequirements groups MAY be included.

The Modify operation can be considered to address cases when a flow is slightly changed, e.g., only MaxPayloadSize (Section 5.5) has been changed. The advantage of having a Modify is that it allows initiation of a change of flow spec while leaving the current flow is operating until the change is accepted. If there is no linkage between the Join and the Leave, then while figuring out whether the new flow spec can be supported, the controller entity has to assume that the resources committed to the current flow are in use. By using Modify the controller entity knows that the resources supporting the current flow can be available for supporting the altered flow. Modify is considered to be an optional operation due to possible controller plane limitations.

8. Summary

This document describes DetNet flow information model and service information model for DetNet IP networks and DetNet MPLS networks.

9. IANA Considerations

N/A.

10. Security Considerations

Security considerations for DetNet are described in detail in [I-D.ietf-detnet-security]. General security considerations are described in [I-D.ietf-detnet-architecture]. This section covers security for the Flow Information Model and there are no additional security considerations introduced by this document.

11. References

11.1. Normative References

[I-D.ietf-detnet-architecture]

Finn, N., Thubert, P., Varga, B., and J. Farkas,
"Deterministic Networking Architecture", draft-ietf-
detnet-architecture-13 (work in progress), May 2019.

[I-D.ietf-detnet-ip]

Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A.,
Bryant, S., and J. Korhonen, "DetNet Data Plane: IP",
draft-ietf-detnet-ip-01 (work in progress), July 2019.

[I-D.ietf-detnet-mpls]

Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A.,
Bryant, S., and J. Korhonen, "DetNet Data Plane: MPLS",
draft-ietf-detnet-mpls-01 (work in progress), July 2019.

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

11.2. Informative References

- [I-D.ietf-detnet-data-plane-framework]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane Framework", draft-ietf-detnet-data-plane-framework-02 (work in progress), September 2019.
- [I-D.ietf-detnet-security]
Mizrahi, T., Grossman, E., Hacker, A., Das, S., Dowdell, J., Austad, H., Stanton, K., and N. Finn, "Deterministic Networking (DetNet) Security Considerations", draft-ietf-detnet-security-05 (work in progress), August 2019.
- [IEEE8021CB]
IEEE Standards Association, "IEEE Std 802.1CB-2017 IEEE Standard for Local and metropolitan area networks - Frame Replication and Elimination for Reliability", 2017, <<https://ieeexplore.ieee.org/document/8091139/>>.
- [IEEE8021Q]
IEEE Standards Association, "IEEE Std 802.1Q-2018 IEEE Standard for Local and metropolitan area networks - Bridges and Bridged Networks", 2018, <<https://ieeexplore.ieee.org/document/8403927/>>.
- [IEEE8021Qbv]
IEEE Standards Association, "IEEE Std 802.1Qbv-2015 IEEE Standard for Local and metropolitan area networks - Bridges and Bridged Networks - Amendment 25: Enhancements for Scheduled Traffic", 2015, <<https://ieeexplore.ieee.org/document/7572858/>>.

[IEEE8021Qcc]

IEEE Standards Association, "IEEE Std 802.1Qcc-2018: IEEE Standard for Local and metropolitan area networks - Bridges and Bridged Networks -- Amendment 31: Stream Reservation Protocol (SRP) Enhancements and Performance Improvements", 2018,
<<https://ieeexplore.ieee.org/document/8514112/>>.

[IETFDetNet]

IETF, "IETF Deterministic Networking (DetNet) Working Group", <<https://datatracker.ietf.org/wg/detnet/charter/>>.

[RFC3444] Pras, A. and J. Schoenwaelder, "On the Difference between Information Models and Data Models", RFC 3444, DOI 10.17487/RFC3444, January 2003,
<<https://www.rfc-editor.org/info/rfc3444>>.

Authors' Addresses

Janos Farkas
Ericsson
Magyar tudosok korutja 11
Budapest 1117
Hungary

Email: janos.farkas@ericsson.com

Balazs Varga
Ericsson
Magyar tudosok korutja 11
Budapest 1117
Hungary

Email: balazs.a.varga@ericsson.com

Rodney Cummings
National Instruments
11500 N. Mopac Expwy
Bldg. C
Austin, TX 78759-3504
USA

Email: rodney.cummings@ni.com

Yuanlong Jiang
Huawei Technologies Co., Ltd.
Bantian, Longgang district
Shenzhen 518129
China

Email: jiangyuanlong@huawei.com

Don Fedyk
LabN Consulting, L.L.C.

Email: dfedyk@labn.net

DetNet
Internet-Draft
Intended status: Standards Track
Expires: April 29, 2020

B. Varga, Ed.
J. Farkas
Ericsson
L. Berger
D. Fedyk
LabN Consulting, L.L.C.
A. Malis
Independent
S. Bryant
Futurewei Technologies
J. Korhonen
October 27, 2019

DetNet Data Plane: IP
draft-ietf-detnet-ip-03

Abstract

This document specifies the Deterministic Networking data plane when operating in an IP packet switched network.

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 April 29, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents

carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Terminology	3
2.1. Terms Used In This Document	3
2.2. Abbreviations	3
2.3. Requirements Language	4
3. DetNet IP Data Plane Overview	4
4. DetNet IP Data Plane Considerations	6
4.1. End-system-specific Considerations	7
4.2. DetNet Domain-Specific Considerations	7
4.3. Forwarding Sub-Layer Considerations	9
4.3.1. Class of Service	9
4.3.2. Quality of Service	10
4.3.3. Path Selection	10
4.4. DetNet Flow Aggregation	11
4.5. Bidirectional Traffic	12
5. DetNet IP Data Plane Procedures	12
5.1. DetNet IP Flow Identification Procedures	12
5.1.1. IP Header Information	13
5.1.2. Other Protocol Header Information	14
5.2. Forwarding Procedures	15
5.3. DetNet IP Traffic Treatment Procedures	15
6. Management and Control Information Summary	16
7. Security Considerations	17
8. IANA Considerations	18
9. Acknowledgements	18
10. References	18
10.1. Normative references	18
10.2. Informative references	19
Authors' Addresses	22

1. Introduction

Deterministic Networking (DetNet) is a service that can be offered by a network to DetNet flows. DetNet provides these flows extremely low packet loss rates and assured maximum end-to-end delivery latency. General background and concepts of DetNet can be found in the DetNet Architecture [I-D.ietf-detnet-architecture].

This document specifies the DetNet data plane operation for IP hosts and routers that provide DetNet service to IP encapsulated data. No

DetNet-specific encapsulation is defined to support IP flows, instead the existing IP and higher layer protocol header information is used to support flow identification and DetNet service delivery. Common data plane procedures and control information for all DetNet data planes can be found in the [I-D.ietf-detnet-data-plane-framework].

The DetNet Architecture models the DetNet related data plane functions as two sub-layers: functions into two sub-layers: a service sub-layer and a forwarding sub-layer. The service sub-layer is used to provide DetNet service protection (e.g., by packet replication and packet elimination functions) and reordering. The forwarding sub-layer is used to provides congestion protection (low loss, assured latency, and limited out-of-order delivery). The service sub-layer generally requires additional fields to provide its service; for example see [I-D.ietf-detnet-mpls]. Since no DetNet-specific fields are added to support DetNet IP flows, only the forwarding sub-layer functions are supported using the DetNet IP defined by this document. Service protection can be provided on a per sub-net basis using technologies such as MPLS [I-D.ietf-detnet-dp-sol-mpls] and Ethernet as specified in the IEEE 802.1 TSN task group (referred to in this document simply as IEEE802.1 TSN).

This document provides an overview of the DetNet IP data plane in Section 3, considerations that apply to providing DetNet services via the DetNet IP data plane in Section 4. Section 5 provides the procedures for hosts and routers that support IP-based DetNet services. Section 6 summarizes the set of information that is needed to identify an individual DetNet flow.

2. Terminology

2.1. Terms Used In This Document

This document uses the terminology and concepts established in the DetNet architecture [I-D.ietf-detnet-architecture], and the reader is assumed to be familiar with that document and its terminology.

2.2. Abbreviations

The following abbreviations used in this document:

CoS	Class of Service.
DetNet	Deterministic Networking.
DN	DetNet.
DiffServ	Differentiated Services

DSCP	Differentiated Services Code Point
L2	Layer-2.
L3	Layer-3.
LSP	Label-switched path.
MPLS	Multiprotocol Label Switching.
PREOF	Packet Replication, Elimination and Ordering Function.
QoS	Quality of Service.
TSN	Time-Sensitive Networking, TSN is a Task Group of the IEEE 802.1 Working Group.

2.3. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. DetNet IP Data Plane Overview

This document describes how IP is used by DetNet nodes, i.e., hosts and routers, to identify DetNet flows and provide a DetNet service using an IP data plane. From a data plane perspective, an end-to-end IP model is followed. As mentioned above, existing IP and higher layer protocol header information is used to support flow identification and DetNet service delivery. Common data plane procedures and control information for all DetNet data planes can be found in the [I-D.ietf-detnet-data-plane-framework].

The DetNet IP data plane primarily uses "6-tuple" based flow identification, where 6-tuple refers to information carried in IP and higher layer protocol headers. The 6-tuple referred to in this document is the same as that defined in [RFC3290]. Specifically 6-tuple is (destination address, source address, IP protocol, source port, destination port, and differentiated services (DiffServ) code point (DSCP). General background on the use of IP headers, and 5-tuples, to identify flows and support Quality of Service (QoS) can be found in [RFC3670]. [RFC7657] also provides useful background on the delivery of DiffServ and "tuple" based flow identification.

The DetNet IP data plane also allows for optional matching on the IPv6 flow label field, as defined in [RFC8200].

Non-DetNet and DetNet IP packets are identical on the wire. Generally the fields used in flow identification are forwarded unmodified, however modification of these fields is allowed, for example to a DSCP value, when required by the DetNet service.

DetNet flow aggregation may be enabled via the use of wildcards, masks, lists, prefixes and ranges. IP tunnels may also be used to support flow aggregation. In these cases, it is expected that DetNet-aware intermediate nodes will provide DetNet service on the aggregate through resource allocation and congestion control mechanisms.

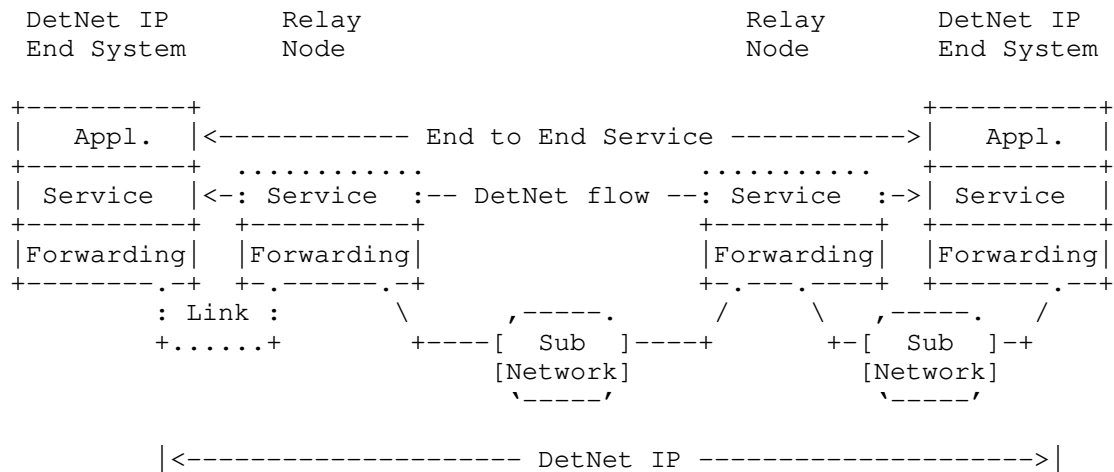


Figure 1: A Simple DetNet (DN) Enabled IP Network

Figure 1 illustrates a DetNet enabled IP network. The DetNet enabled end systems originate IP encapsulated traffic those are identified within the DetNet domain as DetNet flows, relay nodes understand the forwarding requirements of the DetNet flow and ensure that node, interface and sub-network resources are allocated to ensure DetNet service requirements. The dotted line around the Service component of the Relay Nodes indicates that the transit routers are DetNet service aware but do not perform any DetNet service sub-layer function, e.g., PREOF.

Note: The sub-network can represent a TSN, MPLS or IP network segment.

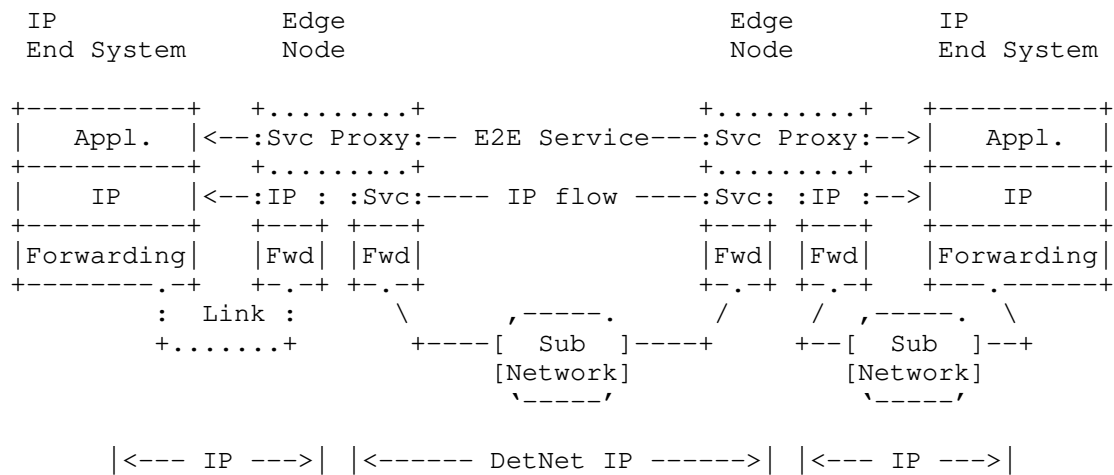


Figure 2: Non-DetNet-aware IP end systems with DetNet IP Domain

Figure 2 illustrates a variant of Figure 1 where the end systems are not DetNet aware. In this case, edge nodes sit at the boundary of the DetNet domain and provide DetNet service proxies for the end applications by initiating and terminating DetNet service for the application's IP flows. The existing header information or an approach such as described in Section 4.4 can be used to support DetNet flow identification.

Note, that Figure 1 and Figure 2 can be collapsed, so IP DetNet End Systems can communicate over DetNet IP network with IP End System.

As non-DetNet and DetNet IP packets are identical on the wire, from data plane perspective, the only difference is that there is flow-associated DetNet information on each DetNet node that defines the flow related characteristics and required forwarding behavior. As shown above, edge nodes provide a Service Proxy function that "associates" one or more IP flows with the appropriate DetNet flow-specific information and ensures that the receives the proper traffic treatment within the domain.

Note: The operation of IEEE802.1 TSN end systems over DetNet enabled IP networks is not described in this document. TSN over MPLS is discribed in [I-D.ietf-detnet-tsn-vpn-over-mpls].

4. DetNet IP Data Plane Considerations

This section provides informative considerations related to providing DetNet service to flows which are identified based on their header information.

4.1. End-system-specific Considerations

Data-flows requiring DetNet service are generated and terminated on end systems. This document deals only with IP end systems. The protocols used by an IP end system are specific to an application, and end systems peer with other end systems. DetNet's use of 6-tuple IP flow identification means that DetNet must be aware of not only the format of the IP header, but also of the next protocol carried within an IP packet (see Section 5.1.1.3).

When IP end systems are DetNet-aware, no application-level or service-level proxy functions are needed inside the DetNet domain. For DetNet unaware IP end systems service-level proxy functions are needed inside the DetNet domain.

End systems need to ensure that DetNet service requirements are met when processing packets associated to a DetNet flow. When forwarding packets, this means that packets are appropriately shaped on transmission and receive appropriate traffic treatment on the connected sub-network, see Section 4.3.2 and Section 4.2 for more details. When receiving packets, this means that there are appropriate local node resources, e.g., buffers, to receive and process the packets of that DetNet flow.

In order to maximize reuse of 5-tuple based mechanisms, e.g., traceroute, DetNet-aware applications and end systems SHOULD NOT mix DetNet and non-DetNet traffic within a single 5-tuple.

4.2. DetNet Domain-Specific Considerations

As a general rule, DetNet IP domains need to be able to forward any DetNet flow identified by the IP 6-tuple. Doing otherwise would limit the number of 6-tuple flow ID combinations that could be used by the end systems. From a practical standpoint this means that all nodes along the end-to-end path of DetNet flows need to agree on what fields are used for flow identification, and the transport protocols (e.g., TCP/UDP/IPsec) which can be used to identify 6-tuple protocol ports.

From a connection type perspective two scenarios are identified:

1. DN attached: the end system is directly connected to an edge node, or the end system is behind a sub-network (See ES1 and ES2 in figure below)
2. DN integrated: the end system is part of the DetNet domain. (See ES3 in figure below)

L3 (IP) end systems may use any of these connection types. A DetNet domain allows communication between any end-systems using the same encapsulation format, independent of their connection type and DetNet capability. DN attached end systems have no knowledge about the DetNet domain and its encapsulation format. See Figure 3 for L3 end system connection examples.

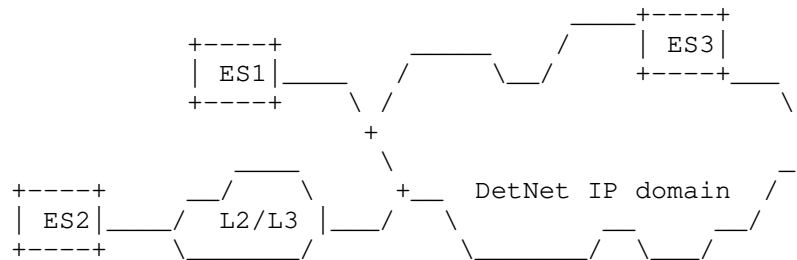


Figure 3: Connection types of L3 end systems

Within a DetNet domain, the DetNet-enabled IP Routers are interconnected by links and sub-networks to support end-to-end delivery of DetNet flows. From a DetNet architecture perspective, these routers are DetNet relays, as they must be DetNet service aware. Such routers identify DetNet flows based on the IP 6-tuple, and ensure that the DetNet service required traffic treatment is provided both on the node and on any attached sub-network.

This solution provides DetNet functions end to end, but does so on a per link and sub-network basis. Congestion protection and latency control and the resource allocation (queuing, policing, shaping) are supported using the underlying link / sub net specific mechanisms. However, service protection (packet replication and packet elimination functions) is not provided at the DetNet layer end to end. Instead service protection can be provided on a per underlying L2 link and sub-network basis.

The DetNet Service Flow is mapped to the link / sub-network specific resources using an underlying system-specific means. This implies each DetNet-aware node on path looks into the forwarded DetNet Service Flow packet and utilize e.g., a 6-tuple to find out the required mapping within a node.

As noted earlier, service protection must be implemented within each link / sub-network independently, using the domain specific mechanisms. This is due to the lack of unified end-to-end sequencing information that could be used by the intermediate nodes. Therefore,

service protection (if enabled) cannot be provided end-to-end, only within sub-networks. This is shown for a three sub-network scenario in Figure 4, where each sub-network can provide service protection between its borders. "R" and "E" denotes replication and elimination points within the sub-network.

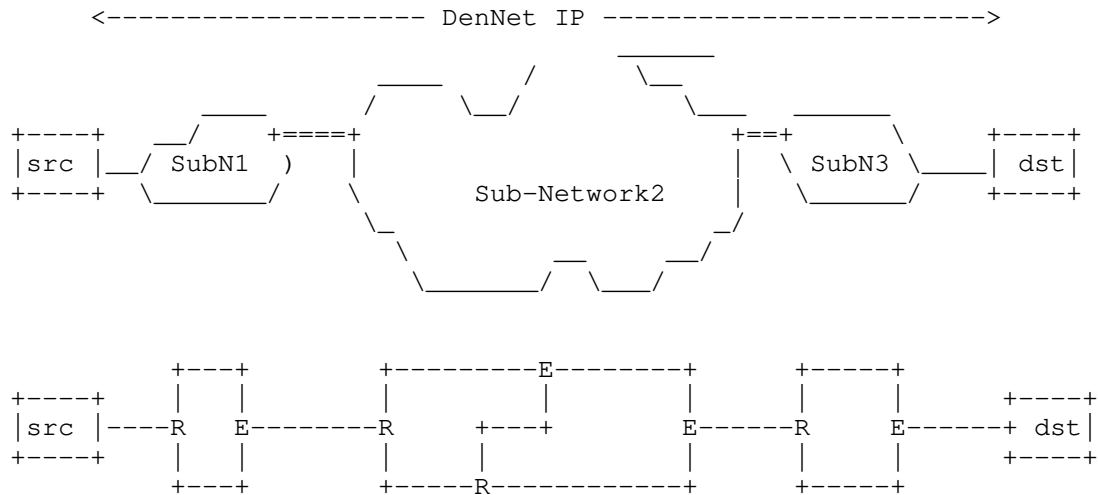


Figure 4: Replication and elimination in sub-networks for DetNet IP networks

If end to end service protection is desired, it can be implemented, for example, by the DetNet end systems using Layer-4 (L4) transport protocols or application protocols. However, these protocols are out of scope of this document.

Note that not mixing DetNet and non-DetNet traffic within a single 5-tuple, as described above, enables simpler 5-tuple filters to be used (or re-used) at the edges of a DetNet network to prevent non-congestion-responsive DetNet traffic from escaping the DetNet domain.

4.3. Forwarding Sub-Layer Considerations

4.3.1. Class of Service

Class of Service (CoS) for DetNet flows carried in IPv6 is provided using the standard differentiated services code point (DSCP) field [RFC2474] and related mechanisms.

One additional consideration for DetNet nodes which support CoS services is that they MUST ensure that the CoS service classes do not impact the congestion protection and latency control mechanisms used to provide DetNet QoS. This requirement is similar to the requirement for MPLS LSRs that CoS LSPs cannot impact the resources allocated to TE LSPs [RFC3473].

4.3.2. Quality of Service

Quality of Service (QoS) for DetNet service flows carried in IP MUST be provided locally by the DetNet-aware hosts and routers supporting DetNet flows. Such support leverages the underlying network layer such as 802.1 TSN. The traffic control mechanisms used to deliver QoS for IP encapsulated DetNet flows are expected to be defined in a future document. From an encapsulation perspective, the combination of the 6-tuple i.e., the typical 5-tuple enhanced with the DSCP and previously mentioned optional field, uniquely identifies a DetNet IP flow.

Packets that are identified as part of a DetNet IP flow but that have not been the subject of a completed reservation, can disrupt the QoS offered to properly reserved DetNet flows by using resources allocated to the reserved flows. Therefore, the network nodes of a DetNet network MUST ensure that no DetNet allocated resources, e.g., queue or shaper, is used by such flows. There are multiple methods that MAY be used by an implementation to defend service delivery to reserved DetNet flows, including but not limited to:

- o Treating packets associated with an incomplete reservation as non-DetNet traffic.
- o Discarding packets associated with an incomplete reservation.
- o Remarking packets associated with an incomplete reservation. Remarking can be accomplished by changing the value of the DSCP, or optional, field to a value that results in the packet no longer matching any other reserved DetNet IP flow.

4.3.3. Path Selection

While path selection algorithms and mechanisms are out of scope of the DetNet data plane definition, it is important to highlight the implications of DetNet IP flow identification on path selection and next hops. As mentioned above, the DetNet IP data plane identifies flows using "6-tuple" header information as well as the additional optional header field. DetNet generally allows for both flow-specific traffic treatment and flow-specific next-hops.

In non-DetNet IP forwarding, it is generally assumed that the same series of next hops, i.e., the same path, will be used for a particular 5-tuple or, in some cases, e.g., [RFC5120], for a particular 6-tuple. Using different next hops for different 5-tuples does not take any special consideration for DetNet-aware applications.

Care should be taken when using different next hops for the same 5-tuple. As discussed in [RFC7657], unexpected behavior can occur when a single 5-tuple application flow experience reordering due to being split across multiple next hops. Understanding of the application and transport protocol impact of using different next hops for the same 6-tuple is required. Again, this impacts path selection for DetNet flows and this document only indirectly.

4.4. DetNet Flow Aggregation

As described in [I-D.ietf-detnet-data-plane-framework], the ability to aggregate individual flows, and their associated resource control, into a larger aggregate is an important technique for improving scaling by reducing the state per hop. DetNet IP data plane aggregation can take place within a single node, when that node maintains state about both the aggregated and individual flows. It can also take place between nodes, where one node maintains state about only flow aggregates while the other node maintains state on all or a portion of the component flows. In either case, the management or control function that provisions the aggregate flows must ensure that adequate resources are allocated and configured to provide combined service requirements of the individual flows. As DetNet is concerned about latency and jitter, more than just bandwidth needs to be considered.

From a single node perspective, the aggregation of IP flows impacts DetNet IP data plane flow identification and resource allocation. As discussed above, IP flow identification uses the IP "6-tuple" for flow identification. DetNet IP flows can be aggregated using any of the 6-tuple, and an additional optional field defined in Section 5.1. The use of prefixes, wildcards, lists, and value ranges allows a DetNet node to identify aggregate DetNet flows. From a resource allocation perspective, DetNet nodes must provide service to an aggregate and not on a component flow basis.

It is the responsibility of the DetNet controller plane to properly provision the use of these aggregation mechanisms. This includes ensuring that aggregated flows have compatible e.g., the same or very similar QoS and/or CoS characteristics, see Section 4.3.2. It also includes ensuring that per component-flow service requirements are satisfied by the aggregate, see Section 5.3.

4.5. Bidirectional Traffic

While the DetNet IP data plane must support bidirectional DetNet flows, there are no special bidirectional features with respect to the data plane other than the need for the two directions of a co-routed bidirectional flow to take the same path. That is to say that bidirectional DetNet flows are solely represented at the management and control plane levels, without specific support or knowledge within the DetNet data plane. Fate sharing and associated or co-routed bidirectional flows can be managed at the control level.

Control and management mechanisms need to support bidirectional flows, but the specification of such mechanisms are out of scope of this document. An example control plane solution for MPLS can be found in [RFC7551].

5. DetNet IP Data Plane Procedures

This section provides DetNet IP data plane procedures. These procedures have been divided into the following areas: flow identification, forwarding and traffic treatment. Flow identification includes those procedures related to matching IP and higher layer protocol header information to DetNet flow (state) information and service requirements. Flow identification is also sometimes called Traffic classification, for example see [RFC5777]. Forwarding includes those procedures related to next hop selection and delivery. Traffic treatment includes those procedures related to providing an identified flow with the required DetNet service.

DetNet IP data plane establishment and operational procedures also have requirements on the control and management systems for DetNet flows and these are referred in this section. Specifically this section identifies a number of information elements that require support via the management and control interfaces supported by a DetNet node. The specific mechanism used for such support is out of the scope of this document. A summary of the requirements for management and control related information is included. Conformance language is not used in the summary since applies to future mechanisms such as those that may be provided in YANG models [I-D.ietf-detnet-yang].

5.1. DetNet IP Flow Identification Procedures

IP and higher layer protocol header information is used to identify DetNet flows. All DetNet implementations that support this document MUST identify individual DetNet flows based on the set of information identified in this section. Note, that additional flow

identification requirements, e.g., to support other higher layer protocols, may be defined in the future.

The configuration and control information used to identify an individual DetNet flow MUST be ordered by an implementation. Implementations MUST support a fixed order when identifying flows, and MUST identify a DetNet flow by the first set of matching flow information.

Implementations of this document MUST support DetNet flow identification when the implementation is acting as a DetNet end systems, a relay node, or as an edge node.

5.1.1.1. IP Header Information

Implementations of this document MUST support DetNet flow identification based on IP header information. The IPv4 header is defined in [RFC0791] and the IPv6 is defined in [RFC8200].

5.1.1.1.1. Source Address Field

Implementations of this document MUST support DetNet flow identification based on the Source Address field of an IP packet. Implementations SHOULD support longest prefix matching for this field, see [RFC1812] and [RFC7608]. Note that a prefix length of zero (0) effectively means that the field is ignored.

5.1.1.1.2. Destination Address Field

Implementations of this document MUST support DetNet flow identification based on the Destination Address field of an IP packet. Implementations SHOULD support longest prefix matching for this field, see [RFC1812] and [RFC7608]. Note that a prefix length of zero (0) effectively means that the field is ignored.

Note: any IP address value is allowed, including an IP multicast destination address.

5.1.1.1.3. IPv4 Protocol and IPv6 Next Header Fields

Implementations of this document MUST support DetNet flow identification based on the IPv4 Protocol field when processing IPv4 packets, and the IPv6 Next Header Field when processing IPv6 packets. An implementation MUST support flow identification based on the next protocol values defined in Section 5.1.2. Other, non-zero values, MUST be used for flow identification. Implementations SHOULD allow for these fields to be ignored for a specific DetNet flow.

5.1.1.4. IPv4 Type of Service and IPv6 Traffic Class Fields

These fields are used to support Differentiated Services [RFC2474] [RFC2475]. Implementations of this document MUST support DetNet flow identification based on the DSCP field in the IPv4 Type of Service field when processing IPv4 packets, and the DSCP field in the IPv6 Traffic Class Field when processing IPv6 packets. Implementations MUST support list based matching of DSCP values, where the list is composed of possible field values that are to be considered when identifying a specific DetNet flow. Implementations SHOULD allow for this field to be ignored for a specific DetNet flow.

5.1.1.5. IPv6 Flow Label Field

Implementations of this document SHOULD support identification of DetNet flows based on the IPv6 Flow Label field. Implementations that support matching based on this field MUST allow for this field to be ignored for a specific DetNet flow. When this field is used to identify a specific DetNet flow, implementations MAY exclude the IPv6 Next Header field and next header information as part of DetNet flow identification.

5.1.2. Other Protocol Header Information

Implementations of this document MUST support DetNet flow identification based on header information identified in this section. Support for TCP, UDP and IPsec flows is defined. Future documents are expected to define support for other protocols.

5.1.2.1. TCP and UDP

DetNet flow identification for TCP [RFC0793] and UDP [RFC0768] is achieved based on the Source and Destination Port fields carried in each protocol's header. These fields share a common format and common DetNet flow identification procedures.

5.1.2.1.1. Source Port Field

Implementations of this document MUST support DetNet flow identification based on the Source Port field of a TCP or UDP packet. Implementations MUST support flow identification based on a particular value carried in the field, i.e., an exact value. Implementations SHOULD support range-based port matching. Implementation MUST also allow for the field to be ignored for a specific DetNet flow.

5.1.2.1.2. Destination Port Field

Implementations of this document MUST support DetNet flow identification based on the Destination Port field of a TCP or UDP packet. Implementations MUST support flow identification based on a particular value carried in the field, i.e., an exact value. Implementations SHOULD support range-based port matching. Implementation MUST also allow for the field to be ignored for a specific DetNet flow.

5.1.2.2. IPsec AH and ESP

IPsec Authentication Header (AH) [RFC4302] and Encapsulating Security Payload (ESP) [RFC4303] share a common format for the Security Parameters Index (SPI) field. Implementations MUST support flow identification based on a particular value carried in the field, i.e., an exact value. Implementation SHOULD also allow for the field to be ignored for a specific DetNet flow.

5.2. Forwarding Procedures

General requirements for IP nodes are defined in [RFC1122], [RFC1812] and [RFC6434], and are not modified by this document. The typical next-hop selection process is impacted by DetNet. Specifically, implementations of this document SHALL use management and control information to select the one or more outgoing interfaces and next hops to be used for a packet associated with a DetNet flow.

The use of multiple paths or links, e.g., ECMP, to support a single DetNet flow is NOT RECOMMENDED. ECMP MAY be used for non-DetNet flows within a DetNet domain.

The above implies that management and control functions will be defined to support this requirement, e.g., see [I-D.ietf-detnet-yang].

5.3. DetNet IP Traffic Treatment Procedures

Implementations of this document MUST ensure that a DetNet flow receives the traffic treatment that is provisioned for it via configuration or the controller plane, e.g., via [I-D.ietf-detnet-yang]. General information on DetNet service can be found in [I-D.ietf-detnet-flow-information-model]. Typical mechanisms used to provide different treatment to different flows includes the allocation of system resources (such as queues and buffers) and provisioning or related parameters (such as shaping, and policing). Support can also be provided via an underlying network technology such as MPLS [I-D.ietf-detnet-ip-over-mpls] or IEEE802.1

TSN [I-D.ietf-detnet-ip-over-tsn]. Other than in the TSN case, the specific mechanisms used by a DetNet node to ensure DetNet service delivery requirements are met for supported DetNet flows is outside the scope of this document.

6. Management and Control Information Summary

The following summarizes the set of information that is needed to identify individual and aggregated DetNet flows:

- o IPv4 and IPv6 source address field.
- o IPv4 and IPv6 source address prefix length, where a zero (0) value effectively means that the address field is ignored.
- o IPv4 and IPv6 destination address field.
- o IPv4 and IPv6 destination address prefix length, where a zero (0) effectively means that the address field is ignored.
- o IPv4 protocol field. A limited set of values is allowed, and the ability to ignore this field, e.g., via configuration of the value zero (0), is desirable.
- o IPv6 next header field. A limited set of values is allowed, and the ability to ignore this field, e.g., via configuration of the value zero (0), is desirable.
- o For the IPv4 Type of Service and IPv6 Traffic Class Fields:
 - * If the DSCP field is to be used in flow identification. Ignoring the DSCP field is optional.
 - * When the DSCP field is used in flow identification, a list of field values that may be used by a specific flow.
- o IPv6 flow label field. This field can be optionally used for matching. When used, can be used instead of matching against the Next Header field.
- o TCP and UDP Source Port. Exact and wildcard matching is required. Port ranges can optionally be used.
- o TCP and UDP Destination Port. Exact and wildcard matching is required. Port ranges can optionally be used.
- o IPsec Header SPI field. Exact matching is required.

This information MUST be provisioned per DetNet flow via configuration, e.g., via the controller or management plane.

Information identifying a DetNet flow is ordered and implementations use the first match. This can, for example, be used to provide a DetNet service for a specific UDP flow, with unique Source and Destination Port field values, while providing a different service for the aggregate of all other flows with that same UDP Destination Port value.

It is the responsibility of the DetNet controller plane to properly provision both flow identification information and the flow specific resources needed to provided the traffic treatment needed to meet each flow's service requirements. This applies for aggregated and individual flows.

7. Security Considerations

Security considerations for DetNet are described in detail in [I-D.ietf-detnet-security]. General security considerations are described in [I-D.ietf-detnet-architecture]. This section considers exclusively security considerations which are specific to the DetNet IP data plane.

Security aspects which are unique to DetNet are those whose aim is to provide the specific quality of service aspects of DetNet, which are primarily to deliver data flows with extremely low packet loss rates and bounded end-to-end delivery latency.

The primary considerations for the data plane is to maintain integrity of data and delivery of the associated DetNet service traversing the DetNet network. Application flows can be protected through whatever means is provided by the underlying technology. For example, encryption may be used, such as that provided by IPsec [RFC4301] for IP flows and/or by an underlying sub-net using MACSec [IEEE802.1AE-2018] for IP over Ethernet (Layer-2) flows.

From a data plane perspective this document does not add or modify any header information.

At the management and control level DetNet flows are identified on a per-flow basis, which may provide controller plane attackers with additional information about the data flows (when compared to controller planes that do not include per-flow identification). This is an inherent property of DetNet which has security implications that should be considered when determining if DetNet is a suitable technology for any given use case.

To provide uninterrupted availability of the DetNet service, provisions can be made against DOS attacks and delay attacks. To protect against DOS attacks, excess traffic due to malicious or malfunctioning devices can be prevented or mitigated, for example through the use of existing mechanism such as policing and shaping applied at the input of a DetNet domain. To prevent DetNet packets from being delayed by an entity external to a DetNet domain, DetNet technology definition can allow for the mitigation of Man-In-The-Middle attacks, for example through use of authentication and authorization of devices within the DetNet domain.

8. IANA Considerations

This document does not require an action from IANA.

9. Acknowledgements

The authors wish to thank Pat Thaler, Norman Finn, Loa Anderson, David Black, Rodney Cummings, Ethan Grossman, Tal Mizrahi, David Mozes, Craig Gunther, George Swallow, Yuanlong Jiang and Carlos J. Bernardos for their various contributions to this work. David Black served as technical advisor to the DetNet working group during the development of this document and provided many valuable comments.

10. References

10.1. Normative references

- [RFC0768] Postel, J., "User Datagram Protocol", STD 6, RFC 768, DOI 10.17487/RFC0768, August 1980, <<https://www.rfc-editor.org/info/rfc768>>.
- [RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, DOI 10.17487/RFC0791, September 1981, <<https://www.rfc-editor.org/info/rfc791>>.
- [RFC0793] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, DOI 10.17487/RFC0793, September 1981, <<https://www.rfc-editor.org/info/rfc793>>.
- [RFC1812] Baker, F., Ed., "Requirements for IP Version 4 Routers", RFC 1812, DOI 10.17487/RFC1812, June 1995, <<https://www.rfc-editor.org/info/rfc1812>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.

- [RFC2474] Nichols, K., Blake, S., Baker, F., and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, DOI 10.17487/RFC2474, December 1998, <<https://www.rfc-editor.org/info/rfc2474>>.
- [RFC2475] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., and W. Weiss, "An Architecture for Differentiated Services", RFC 2475, DOI 10.17487/RFC2475, December 1998, <<https://www.rfc-editor.org/info/rfc2475>>.
- [RFC3473] Berger, L., Ed., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions", RFC 3473, DOI 10.17487/RFC3473, January 2003, <<https://www.rfc-editor.org/info/rfc3473>>.
- [RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <<https://www.rfc-editor.org/info/rfc4301>>.
- [RFC4302] Kent, S., "IP Authentication Header", RFC 4302, DOI 10.17487/RFC4302, December 2005, <<https://www.rfc-editor.org/info/rfc4302>>.
- [RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, DOI 10.17487/RFC4303, December 2005, <<https://www.rfc-editor.org/info/rfc4303>>.
- [RFC7608] Boucadair, M., Petrescu, A., and F. Baker, "IPv6 Prefix Length Recommendation for Forwarding", BCP 198, RFC 7608, DOI 10.17487/RFC7608, July 2015, <<https://www.rfc-editor.org/info/rfc7608>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <<https://www.rfc-editor.org/info/rfc8200>>.

10.2. Informative references

- [I-D.ietf-detnet-architecture]
Finn, N., Thubert, P., Varga, B., and J. Farkas,
"Deterministic Networking Architecture", draft-ietf-
detnet-architecture-13 (work in progress), May 2019.
- [I-D.ietf-detnet-data-plane-framework]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A.,
Bryant, S., and J. Korhonen, "DetNet Data Plane
Framework", draft-ietf-detnet-data-plane-framework-02
(work in progress), September 2019.
- [I-D.ietf-detnet-dp-sol-mpls]
Korhonen, J. and B. Varga, "DetNet MPLS Data Plane
Encapsulation", draft-ietf-detnet-dp-sol-mpls-02 (work in
progress), March 2019.
- [I-D.ietf-detnet-flow-information-model]
Farkas, J., Varga, B., Cummings, R., Jiang, Y., and D.
Fedyk, "DetNet Flow Information Model", draft-ietf-detnet-
flow-information-model-05 (work in progress), September
2019.
- [I-D.ietf-detnet-ip-over-mpls]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A.,
Bryant, S., and J. Korhonen, "DetNet Data Plane: IP over
MPLS", draft-ietf-detnet-ip-over-mpls-01 (work in
progress), July 2019.
- [I-D.ietf-detnet-ip-over-tsn]
Varga, B., Farkas, J., Malis, A., Bryant, S., and J.
Korhonen, "DetNet Data Plane: IP over IEEE 802.1 Time
Sensitive Networking (TSN)", draft-ietf-detnet-ip-over-
tsn-00 (work in progress), May 2019.
- [I-D.ietf-detnet-mpls]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A.,
Bryant, S., and J. Korhonen, "DetNet Data Plane: MPLS",
draft-ietf-detnet-mpls-01 (work in progress), July 2019.
- [I-D.ietf-detnet-security]
Mizrahi, T., Grossman, E., Hacker, A., Das, S., Dowdell,
J., Austad, H., Stanton, K., and N. Finn, "Deterministic
Networking (DetNet) Security Considerations", draft-ietf-
detnet-security-05 (work in progress), August 2019.

- [I-D.ietf-detnet-tsn-vpn-over-mpls]
Varga, B., Farkas, J., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: IEEE 802.1 Time Sensitive Networking over MPLS", draft-ietf-detnet-tsn-vpn-over-mpls-00 (work in progress), May 2019.
- [I-D.ietf-detnet-yang]
Geng, X., Chen, M., Ryoo, Y., Li, Z., and R. Rahman, "Deterministic Networking (DetNet) Configuration YANG Model", draft-ietf-detnet-yang-03 (work in progress), July 2019.
- [IEEE802.1AE-2018]
IEEE Standards Association, "IEEE Std 802.1AE-2018 MAC Security (MACsec)", 2018,
<<https://ieeexplore.ieee.org/document/8585421>>.
- [RFC1122] Braden, R., Ed., "Requirements for Internet Hosts - Communication Layers", STD 3, RFC 1122, DOI 10.17487/RFC1122, October 1989,
<<https://www.rfc-editor.org/info/rfc1122>>.
- [RFC3290] Bernet, Y., Blake, S., Grossman, D., and A. Smith, "An Informal Management Model for Diffserv Routers", RFC 3290, DOI 10.17487/RFC3290, May 2002,
<<https://www.rfc-editor.org/info/rfc3290>>.
- [RFC3670] Moore, B., Durham, D., Strassner, J., Westerinen, A., and W. Weiss, "Information Model for Describing Network Device QoS Datapath Mechanisms", RFC 3670, DOI 10.17487/RFC3670, January 2004, <<https://www.rfc-editor.org/info/rfc3670>>.
- [RFC5120] Przygienda, T., Shen, N., and N. Sheth, "M-ISIS: Multi Topology (MT) Routing in Intermediate System to Intermediate Systems (IS-ISs)", RFC 5120, DOI 10.17487/RFC5120, February 2008,
<<https://www.rfc-editor.org/info/rfc5120>>.
- [RFC5777] Korhonen, J., Tschofenig, H., Arumaithurai, M., Jones, M., Ed., and A. Lior, "Traffic Classification and Quality of Service (QoS) Attributes for Diameter", RFC 5777, DOI 10.17487/RFC5777, February 2010,
<<https://www.rfc-editor.org/info/rfc5777>>.
- [RFC6434] Jankiewicz, E., Loughney, J., and T. Narten, "IPv6 Node Requirements", RFC 6434, DOI 10.17487/RFC6434, December 2011, <<https://www.rfc-editor.org/info/rfc6434>>.

- [RFC7551] Zhang, F., Ed., Jing, R., and R. Gandhi, Ed., "RSVP-TE Extensions for Associated Bidirectional Label Switched Paths (LSPs)", RFC 7551, DOI 10.17487/RFC7551, May 2015, <<https://www.rfc-editor.org/info/rfc7551>>.
- [RFC7657] Black, D., Ed. and P. Jones, "Differentiated Services (Diffserv) and Real-Time Communication", RFC 7657, DOI 10.17487/RFC7657, November 2015, <<https://www.rfc-editor.org/info/rfc7657>>.

Authors' Addresses

Balazs Varga (editor)
Ericsson
Magyar Tudosok krt. 11.
Budapest 1117
Hungary

Email: balazs.a.varga@ericsson.com

Janos Farkas
Ericsson
Magyar Tudosok krt. 11.
Budapest 1117
Hungary

Email: janos.farkas@ericsson.com

Lou Berger
LabN Consulting, L.L.C.

Email: lberger@labn.net

Don Fedyk
LabN Consulting, L.L.C.

Email: dfedyk@labn.net

Andrew G. Malis
Independent

Email: agmalis@gmail.com

Stewart Bryant
Futurewei Technologies

Email: stewart.bryant@gmail.com

Jouni Korhonen

Email: jouni.nospam@gmail.com

DetNet
Internet-Draft
Intended status: Standards Track
Expires: April 29, 2020

B. Varga, Ed.
J. Farkas
Ericsson
L. Berger
D. Fedyk
LabN Consulting, L.L.C.
A. Malis
Independent
S. Bryant
Futurewei Technologies
J. Korhonen
October 27, 2019

DetNet Data Plane: IP over MPLS
draft-ietf-detnet-ip-over-mpls-03

Abstract

This document specifies the Deterministic Networking data plane when operating in an IP over MPLS packet switched network.

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 April 29, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents

carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Terminology	2
2.1. Terms Used In This Document	3
2.2. Abbreviations	3
2.3. Requirements Language	3
3. DetNet IP Data Plane Overview	4
4. IP over DetNet MPLS	4
4.1. IP Over DetNet MPLS Data Plane Scenarios	5
4.2. DetNet IP over DetNet MPLS Encapsulation	6
5. IP over DetNet MPLS Procedures	8
5.1. DetNet IP over DetNet MPLS Flow Identification Procedures	8
5.2. DetNet IP over DetNet MPLS Traffic Treatment Procedures .	8
6. Management and Control Information Summary	9
7. Security Considerations	9
8. IANA Considerations	10
9. Acknowledgements	10
10. References	10
10.1. Normative references	10
10.2. Informative references	11
Authors' Addresses	12

1. Introduction

Deterministic Networking (DetNet) is a service that can be offered by a network to DetNet flows. DetNet provides these flows extremely low packet loss rates and assured maximum end-to-end delivery latency. General background and concepts of DetNet can be found in the DetNet Architecture [I-D.ietf-detnet-architecture].

This document specifies use of the IP DetNet encapsulation over an MPLS network. It maps the IP data plane encapsulation described in [I-D.ietf-detnet-ip] to the DetNet MPLS data plane defined in [I-D.ietf-detnet-mpls].

2. Terminology

2.1. Terms Used In This Document

This document uses the terminology and concepts established in the DetNet architecture [I-D.ietf-detnet-architecture] and [I-D.ietf-detnet-data-plane-framework], and the reader is assumed to be familiar with these documents and their terminology.

2.2. Abbreviations

This document uses the abbreviations defined in the DetNet architecture [I-D.ietf-detnet-architecture] and [I-D.ietf-detnet-data-plane-framework]. This document uses the following abbreviations:

CE	Customer Edge equipment.
DetNet	Deterministic Networking.
DF	DetNet Flow.
DN	DetNet.
L2	Layer-2.
L3	Layer-3.
LSP	Label-switched path.
MPLS	Multiprotocol Label Switching.
PE	Provider Edge.
PREOF	Packet Replication, Ordering and Elimination Function.
PSN	Packet Switched Network.
PW	Pseudowire.
TE	Traffic Engineering.
TSN	Time-Sensitive Networking, TSN is a Task Group of the IEEE 802.1 Working Group.

2.3. Requirements Language

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

14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. DetNet IP Data Plane Overview

Figure 1 illustrates an IP DetNet, with an MPLS based DetNet network as a sub-network between the relay nodes. It shows a more complex DetNet enabled IP network where an IP flow is mapped to one or more PWs and MPLS (TE) LSPs. The end systems still originate IP encapsulated traffic that are identified as DetNet flows. The relay nodes follow procedures defined in Section 4 to map each DetNet flow to MPLS LSPs. While not shown, relay nodes can provide service sub-layer functions such as PREOF using DetNet over MPLS, and this is indicated by the solid line for the MPLS facing portion of the Service component. Note that the Transit node is MPLS (TE) LSP aware and performs switching based on MPLS labels, and need not have any specific knowledge of the DetNet service or the corresponding DetNet flow identification. See Section 4 for details on the mapping of IP flows to MPLS, and [I-D.ietf-detnet-mpls] for general support of DetNet services using MPLS.

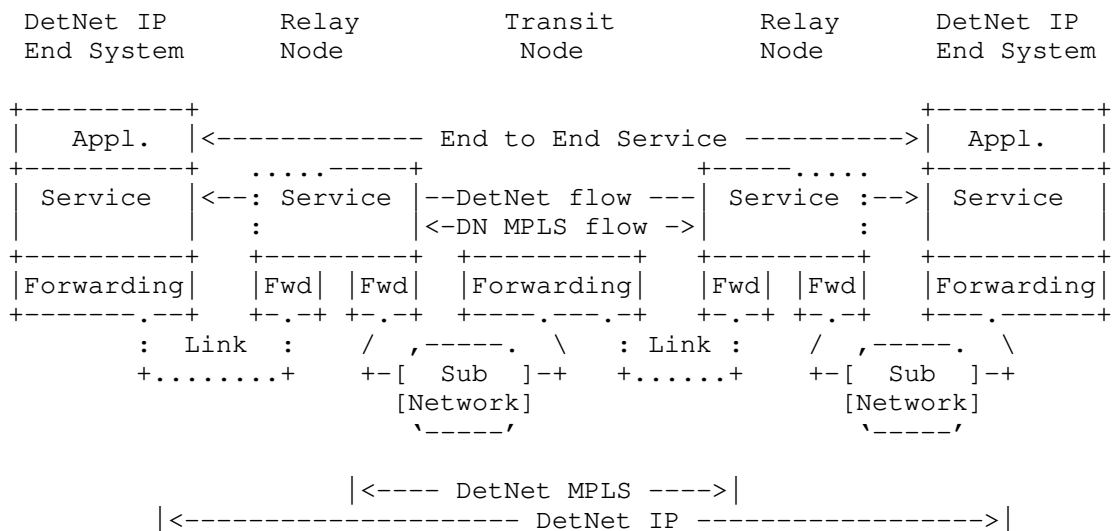


Figure 1: DetNet IP Over DetNet MPLS Network

4. IP over DetNet MPLS

This section defines how IP encapsulated flows are carried over a DetNet MPLS data plane as defined in [I-D.ietf-detnet-mpls]. Since both Non-DetNet and DetNet IP packet are identical on the wire, this

section is applicable to any node that supports IP over DetNet MPLS, and this section refers to both cases as DetNet IP over DetNet MPLS.

4.1. IP Over DetNet MPLS Data Plane Scenarios

An example use of DetNet IP over DetNet MPLS is presented here.

Figure 1 illustrated DetNet enabled End Systems (hosts), connected to DetNet (DN) enabled IP networks, operating over a DetNet aware MPLS network. Using this figure we can have a case where the Relay nodes act as T-PEs and sit at the boundary of the MPLS domain since the non-MPLS domain is DetNet aware. This case is very similar to the DetNet MPLS Network figure 2 in [I-D.ietf-detnet-mpls]. However in [I-D.ietf-detnet-mpls] figure 2 the T-PEs are located at the end system and MPLS spans the whole DetNet service. The primary difference in this document is that the Relay nodes are at the edges of the MPLS domain and therefore function as T-PEs, and that MPLS service sub-layer functions are not provided over the DetNet IP network. The transit node functions show above are identical to those described in [I-D.ietf-detnet-mpls].

Figure 2 illustrates how relay nodes can provide service protection over an MPLS domain. In this case, CE1 and CE2 are IP DetNet end systems which are interconnected via a MPLS domain such as described in [I-D.ietf-detnet-mpls]. Note that R1 and R3 sit at the edges of an MPLS domain and therefore are similar to T-PEs, while R2 sits in the middle of the domain and is therefore similar to an S-PE.

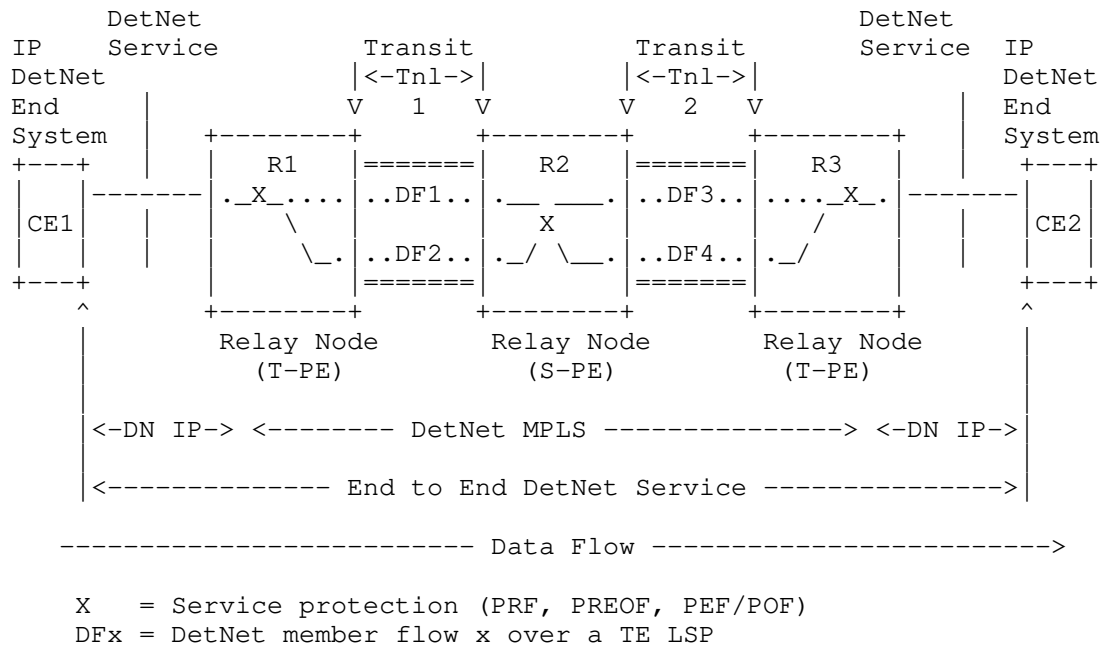


Figure 2: DetNet IP Over DetNet MPLS Network

Figure 1 illustrates DetNet enabled End Systems, connected to DetNet (DN) enabled MPLS network. A similar situation occurs when end systems are not DetNet aware. In this case, edge nodes sit at the boundary of the MPLS domain since it is also a DetNet domain boundary. The edge nodes provide DetNet service proxies for the end applications by initiating and terminating DetNet service for the application's IP flows. While the node types differ, there is essentially no difference in data plane processing between relay and edges. There are likely to be differences in controller plane operation, particularly when distributed control plane protocols are used.

It is still possible to provide DetNet service protection for non-DetNet aware end systems. This case is basically the same as Figure 2, with the exception that CE1 and CE2 are non-DetNet aware end systems and R1 and R3 become edge nodes.

4.2. DetNet IP over DetNet MPLS Encapsulation

The basic encapsulation approach is to treat a DetNet IP flow as an app-flow from the DetNet MPLS perspective. The corresponding example DetNet Sub-Network format is shown in Figure 3.

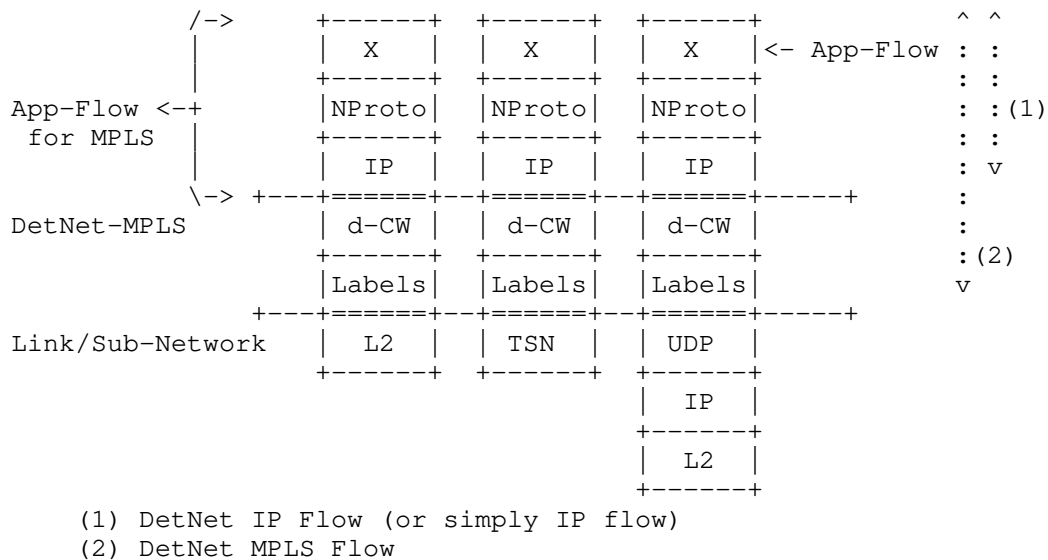


Figure 3: Example DetNet IP over MPLS Sub-Network Formats

In Figure 3 "App-Flow" indicates the payload carried by the DetNet IP data plane. "IP" and "NProto" indicate the fields described in Section 5.1.1. IP Header Information and Section 5.1.2. Other Protocol Header Information in [I-D.ietf-detnet-ip], respectively. "App-Flow for MPLS" indicates that an individual DetNet IP flow is the payload from the perspective of the DetNet MPLS data plane defined in [I-D.ietf-detnet-mpls].

Per [I-D.ietf-detnet-mpls], the DetNet MPLS data plane uses a single S-Label to support a single app flow. Section 5.1. DetNet IP Flow Identification Procedures in [I-D.ietf-detnet-ip] states that a single DetNet flow is identified based on IP, and next level protocol, header information. Section 4.4. Aggregation Considerations in [I-D.ietf-detnet-ip] defines the ways in which aggregation is supported through the use of prefixes, wildcards, lists, and port ranges. Collectively, this results in the fairly straightforward procedures defined in this section.

As shown in Figure 2, DetNet relay nodes are responsible for the mapping of a DetNet flow, at the service sub-layer, from the IP to MPLS DetNet data planes and back again. Their related DetNet IP over DetNet MPLS data plane operation is comprised of two sets of procedures: the mapping of flow identifiers, and ensuring proper traffic treatment.

Mapping of IP to DetNet MPLS is similar for DetNet IP flows and IP flows. The six-tuple of IP is mapped to the S-Label in both cases. The various fields may be mapped or ignored when going from IP to MPLS.

5. IP over DetNet MPLS Procedures

5.1. DetNet IP over DetNet MPLS Flow Identification Procedures

A DetNet relay node (ingress T-PE) that sends a DetNet IP flow over a DetNet MPLS network MUST map a DetNet IP flow, as identified in [I-D.ietf-detnet-ip] into a single MPLS DetNet flow and MUST process it in accordance to the procedures defined in [I-D.ietf-detnet-mpls] Section 6.1. PRF MAY be supported at the MPLS level for DetNet IP flows sent over an DetNet MPLS network. Aggregation MAY be supported as defined in [I-D.ietf-detnet-mpls] Section 5.4. Aggregation considerations in [I-D.ietf-detnet-ip] MAY be used to identify an individual DetNet IP flow. The provisioning of the mapping of DetNet IP flows to DetNet MPLS flows MUST be supported via configuration, e.g., via the controller plane.

A DetNet relay node (egress T-PE) MAY be provisioned to handle packets received via the DetNet MPLS data plane as DetNet IP flows. A single incoming DetNet MPLS flow MAY be treated as a single DetNet IP flow, without examination of IP headers. Alternatively, packets received via the DetNet MPLS data plane MAY follow the normal DetNet IP flow identification procedures defined in [I-D.ietf-detnet-ip] Section 7.1.

An implementation MUST support the provisioning for handling any received DetNet MPLS data plane as DetNet IP flows via configuration. Note that such configuration MAY include support from PREOF on the incoming DetNet MPLS flow.

5.2. DetNet IP over DetNet MPLS Traffic Treatment Procedures

The traffic treatment required for a particular DetNet IP flow is provisioned via configuration or the controller plane. When a DetNet IP flow is sent over DetNet MPLS, a DetNet relay node MUST ensure that the provisioned DetNet IP traffic treatment is provided at the forwarding sub-layer as described in [I-D.ietf-detnet-mpls] Section 5.2. Note that the PRF function MAY be utilized when sending IP over MPLS.

Traffic treatment for DetNet IP flows received over the DetNet MPLS data plane MUST follow Section 5.3 DetNet IP Traffic Treatment Procedures in [I-D.ietf-detnet-ip].

6. Management and Control Information Summary

The following summarizes the set of information that is needed to support DetNet IP over DetNet MPLS at the MPLS ingress node:

- o Each MPLS App-Flow is identified using the IP flow identification information as defined in [I-D.ietf-detnet-ip]. The information is summarized in Section 5.1 of that document, and includes all wildcards, port ranges and the ability to ignore specific IP fields.
- o The DetNet MPLS service that is to be used to send the matching IP traffic. This matching information is provided in [I-D.ietf-detnet-mpls] Section 5.1, and includes both service and traffic delivery information.

The following summarizes the set of information that is needed to support DetNet IP over DetNet MPLS at the MPLS egress node:

- o S-Label values that are carrying MPLS over IP encapsulated traffic.
- o For each S-Label, how the received traffic is to be handled. The traffic may be processed according as any other DetNet IP traffic as defined in this document or in [I-D.ietf-detnet-ip], or the traffic may be directly treated as an MPLS App-flow for additional processing according to [I-D.ietf-detnet-mpls].

It is the responsibility of the DetNet controller plane to properly provision both flow identification information and the flow specific resources needed to provided the traffic treatment needed to meet each flow's service requirements. This applies for aggregated and individual flows.

7. Security Considerations

This draft does not have additional security considerations. Security considerations for DetNet are described in detail in [I-D.ietf-detnet-security]. General security considerations are described in [I-D.ietf-detnet-architecture]. MPLS and IP specific considerations are described in [I-D.ietf-detnet-mpls] and [I-D.ietf-detnet-ip].

Security aspects which are unique to DetNet are those whose aim is to provide the specific quality of service aspects of DetNet, which are primarily to deliver data flows with extremely low packet loss rates and bounded end-to-end delivery latency.

The primary considerations for the data plane is to maintain integrity of data and delivery of the associated DetNet service traversing the DetNet network. Application flows can be protected through whatever means is provided by the underlying technology. For example, encryption may be used, such as that provided by IPSec [RFC4301] for IP flows and/or by an underlying sub-net using MACSec [IEEE802.1AE-2018] for IP over Ethernet (Layer-2) flows.

From a data plane perspective this document does not add or modify any header information.

At the management and control level DetNet flows are identified on a per-flow basis, which may provide controller plane attackers with additional information about the data flows (when compared to controller planes that do not include per-flow identification). This is an inherent property of DetNet which has security implications that should be considered when determining if DetNet is a suitable technology for any given use case.

To provide uninterrupted availability of the DetNet service, provisions can be made against DOS attacks and delay attacks. To protect against DOS attacks, excess traffic due to malicious or malfunctioning devices can be prevented or mitigated, for example through the use of existing mechanism such as policing and shaping applied at the input of a DetNet domain. To prevent DetNet packets from being delayed by an entity external to a DetNet domain, DetNet technology definition can allow for the mitigation of Man-In-The-Middle attacks, for example through use of authentication and authorization of devices within the DetNet domain.

8. IANA Considerations

This document makes no IANA requests.

9. Acknowledgements

The authors wish to thank Pat Thaler, Norman Finn, Loa Anderson, David Black, Rodney Cummings, Ethan Grossman, Tal Mizrahi, David Mozes, Craig Gunther, George Swallow, Yuanlong Jiang and Carlos J. Bernardos for their various contributions to this work.

10. References

10.1. Normative references

- [I-D.ietf-detnet-architecture]
Finn, N., Thubert, P., Varga, B., and J. Farkas,
"Deterministic Networking Architecture", draft-ietf-
detnet-architecture-13 (work in progress), May 2019.
- [I-D.ietf-detnet-ip]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A.,
Bryant, S., and J. Korhonen, "DetNet Data Plane: IP",
draft-ietf-detnet-ip-01 (work in progress), July 2019.
- [I-D.ietf-detnet-mpls]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A.,
Bryant, S., and J. Korhonen, "DetNet Data Plane: MPLS",
draft-ietf-detnet-mpls-01 (work in progress), July 2019.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
Requirement Levels", BCP 14, RFC 2119,
DOI 10.17487/RFC2119, March 1997,
<<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC
2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174,
May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

10.2. Informative references

- [I-D.ietf-detnet-data-plane-framework]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A.,
Bryant, S., and J. Korhonen, "DetNet Data Plane
Framework", draft-ietf-detnet-data-plane-framework-02
(work in progress), September 2019.
- [I-D.ietf-detnet-security]
Mizrahi, T., Grossman, E., Hacker, A., Das, S., Dowdell,
J., Austad, H., Stanton, K., and N. Finn, "Deterministic
Networking (DetNet) Security Considerations", draft-ietf-
detnet-security-05 (work in progress), August 2019.
- [IEEE802.1AE-2018]
IEEE Standards Association, "IEEE Std 802.1AE-2018 MAC
Security (MACsec)", 2018,
<<https://ieeexplore.ieee.org/document/8585421>>.
- [RFC4301] Kent, S. and K. Seo, "Security Architecture for the
Internet Protocol", RFC 4301, DOI 10.17487/RFC4301,
December 2005, <<https://www.rfc-editor.org/info/rfc4301>>.

Authors' Addresses

Balazs Varga (editor)
Ericsson
Magyar Tudosok krt. 11.
Budapest 1117
Hungary

Email: balazs.a.varga@ericsson.com

Janos Farkas
Ericsson
Magyar Tudosok krt. 11.
Budapest 1117
Hungary

Email: janos.farkas@ericsson.com

Lou Berger
LabN Consulting, L.L.C.

Email: lberger@labn.net

Don Fedyk
LabN Consulting, L.L.C.

Email: dfedyk@labn.net

Andrew G. Malis
Independent

Email: agmalis@gmail.com

Stewart Bryant
Futurewei Technologies

Email: stewart.bryant@gmail.com

Jouni Korhonen

Email: jouni.nospam@gmail.com

DetNet
Internet-Draft
Intended status: Standards Track
Expires: April 29, 2020

B. Varga, Ed.
J. Farkas
Ericsson
A. Malis
Independent
S. Bryant
Futurewei Technologies
October 27, 2019

DetNet Data Plane: IP over IEEE 802.1 Time Sensitive Networking (TSN)
draft-ietf-detnet-ip-over-tsn-01

Abstract

This document specifies the Deterministic Networking IP data plane when operating over a TSN sub-network.

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 April 29, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Terminology	3
2.1. Terms Used In This Document	3
2.2. Abbreviations	3
2.3. Requirements Language	3
3. DetNet IP Data Plane Overview	3
4. DetNet IP Flows over an IEEE 802.1 TSN sub-network	5
4.1. Functions for DetNet Flow to TSN Stream Mapping	6
4.2. TSN requirements of IP DetNet nodes	6
4.3. Service protection within the TSN sub-network	8
4.4. Aggregation during DetNet flow to TSN Stream mapping	8
5. Management and Control Implications	8
6. Security Considerations	9
7. IANA Considerations	10
8. Acknowledgements	10
9. References	10
9.1. Normative references	10
9.2. Informative references	10
Authors' Addresses	12

1. Introduction

Deterministic Networking (DetNet) is a service that can be offered by a network to DetNet flows. DetNet provides these flows extremely low packet loss rates and assured maximum end-to-end delivery latency. General background and concepts of DetNet can be found in the DetNet Architecture [I-D.ietf-detnet-architecture].

[I-D.ietf-detnet-ip] specifies the DetNet data plane operation for IP hosts and routers that provide DetNet service to IP encapsulated data. This document focuses on the scenario where DetNet IP nodes are interconnected by a TSN sub-network.

The DetNet Architecture decomposes the DetNet related data plane functions into two sub-layers: a service sub-layer and a forwarding sub-layer. The service sub-layer is used to provide DetNet service protection and reordering. The forwarding sub-layer is used to provides congestion protection (low loss, assured latency, and limited reordering). As described in [I-D.ietf-detnet-ip] no DetNet specific headers are added to support DetNet IP flows, only the forwarding sub-layer functions are supported inside the DetNet domain. Service protection can be provided on a per sub-network basis as shown here for the IEEE802.1 TSN sub-network scenario.

2. Terminology

[Editor's note: Needs clean up.].

2.1. Terms Used In This Document

This document uses the terminology and concepts established in the DetNet architecture [I-D.ietf-detnet-architecture], and the reader is assumed to be familiar with that document and its terminology.

2.2. Abbreviations

The following abbreviations used in this document:

DetNet	Deterministic Networking.
DF	DetNet Flow.
L2	Layer-2.
L3	Layer-3.
PREOF	Packet Replication, Ordering and Elimination Function.
TSN	Time-Sensitive Networking, TSN is a Task Group of the IEEE 802.1 Working Group.

2.3. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. DetNet IP Data Plane Overview

[Editor's note: this section and highlights that DetNet IP over subnets scenario being the focus in the remaining part of the document.].

[I-D.ietf-detnet-ip] describes how IP is used by DetNet nodes, i.e., hosts and routers, to identify DetNet flows and provide a DetNet service. From a data plane perspective, an end-to-end IP model is followed. DetNet uses "6-tuple" based flow identification, where "6-tuple" refers to information carried in IP and higher layer protocol headers.

DetNet flow aggregation may be enabled via the use of wildcards, masks, prefixes and ranges. IP tunnels may also be used to support flow aggregation. In these cases, it is expected that DetNet aware intermediate nodes will provide DetNet service assurance on the aggregate through resource allocation and congestion control mechanisms.

Congestion protection, latency control and the resource allocation (queuing, policing, shaping) are supported using the underlying link / sub-net specific mechanisms. Service protections (packet replication and packet elimination functions) are not provided at the DetNet layer end to end due the lack of a unified end to end sequencing information that would be available for intermediate nodes. However, such service protection can be provided on a per underlying L2 link and sub-network basis.

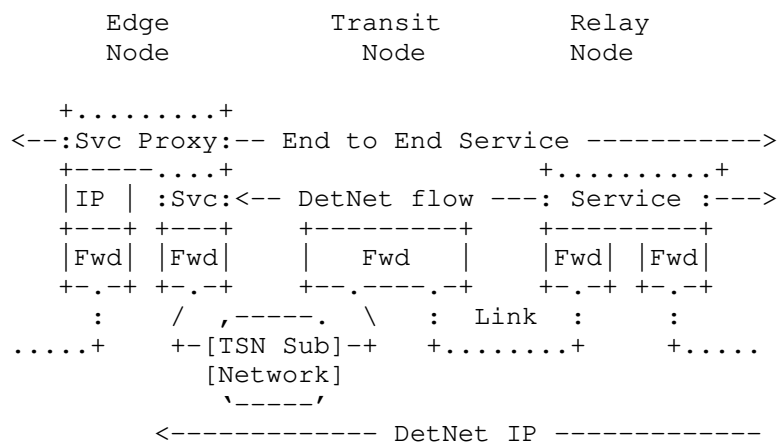


Figure 1: Part of a Simple DetNet (DN) Enabled IP Network using a TSN sub-net

Figure 1 illustrates an extract of a DetNet enabled IP network, that uses a TSN sub-network as interconnection between two DetNet Nodes. In this figure, an Edge Node sits at the boundary of the DetNet domain and provide DetNet service proxies for the end applications by initiating and terminating DetNet service for the application's IP flows. Node and interface resources are allocated to ensure DetNet service requirements. Dotted lines around the Service components of the Edge and Relay Nodes indicate that they are DetNet service aware but do not perform any DetNet service sub-layer function, e.g., PREOF (Packet Replication, Elimination, and Ordering Functions). In this example the Edge Node and the Transit Node are interconnected by a TSN sub-network, being the primary focus of this document.

DetNet routers ensure that detnet service requirements are met per hop by allocating local resources, both receive and transmit, and by mapping the service requirements of each flow to appropriate sub-network mechanisms. Such mappings are sub-network technology specific. The mapping of DetNet IP flows to TSN streams and TSN protection mechanisms are covered in Section 4.

4. DetNet IP Flows over an IEEE 802.1 TSN sub-network

[Authors note: how do we handle control protocols such as ICMP, IPsec, etc.? If such protocols are part of the DetNet flow they can be identified by the Mask-and-match Stream identification function of P802.1CBdb.]

This section covers how DetNet IP flows operate over an IEEE 802.1 TSN sub-network. Figure 2 illustrates such a scenario, where two IP (DetNet) nodes are interconnected by a TSN sub-network. Node-1 is single homed and Node-2 is dual-homed to the TSN sub-network.

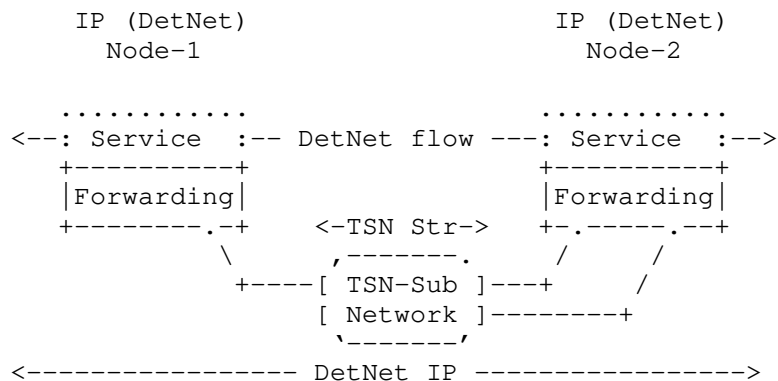


Figure 2: DetNet (DN) Enabled IP Network over a TSN sub-network

The Time-Sensitive Networking (TSN) Task Group of the IEEE 802.1 Working Group have defined (and are defining) a number of amendments to IEEE 802.1Q [IEEE8021Q] that provide zero congestion loss and bounded latency in bridged networks. Furthermore IEEE 802.1CB [IEEE8021CB] defines frame replication and elimination functions for reliability that should prove both compatible with and useful to DetNet networks. All these functions have to identify flows that require TSN treatment.

TSN capabilities of the TSN sub-network are made available for IP (DetNet) flows via the protocol interworking function defined in IEEE 802.1CB [IEEE8021CB]. For example, applied on the TSN edge port it can convert an ingress unicast IP (DetNet) flow to use a specific

Layer-2 multicast destination MAC address and a VLAN, in order to direct the packet through a specific path inside the bridged network. A similar interworking function pair at the other end of the TSN sub-network would restore the packet to its original Layer-2 destination MAC address and VLAN.

Placement of TSN functions depends on the TSN capabilities of nodes. IP (DetNet) Nodes may or may not support TSN functions. For a given TSN Stream (i.e., DetNet flow) an IP (DetNet) node is treated as a Talker or a Listener inside the TSN sub-network.

4.1. Functions for DetNet Flow to TSN Stream Mapping

Mapping of a DetNet IP flow to a TSN Stream is provided via the combination of a passive and an active stream identification function that operate at the frame level. The passive stream identification function is used to catch the 6-tuple of a DetNet IP flow and the active stream identification function is used to modify the Ethernet header according to ID of the mapped TSN Stream.

IEEE 802.1CB [IEEE8021CB] defines an IP Stream identification function that can be used as a passive function for IP DetNet flows using UDP or TCP. IEEE P802.1CBdb [IEEEP8021CBdb] defines a Mask-and-Match Stream identification function that can be used as a passive function for any IP DetNet flows.

IEEE 802.1CB [IEEE8021CB] defines an Active Destination MAC and VLAN Stream identification function, what can replace some Ethernet header fields namely (1) the destination MAC-address, (2) the VLAN-ID and (3) priority parameters with alternate values. Replacement is provided for the frame passed down the stack from the upper layers or up the stack from the lower layers.

Active Destination MAC and VLAN Stream identification can be used within a Talker to set flow identity or a Listener to recover the original addressing information. It can be used also in a TSN bridge that is providing translation as a proxy service for an End System.

4.2. TSN requirements of IP DetNet nodes

This section covers required behavior of a TSN-aware DetNet node using a TSN sub-network.

From the TSN sub-network perspective DetNet IP nodes are treated as Talker or Listener, that may be (1) TSN-unaware or (2) TSN-aware.

In cases of TSN-unaware IP DetNet nodes the TSN relay nodes within the TSN sub-network must modify the Ethernet encapsulation of the

DetNet IP flow (e.g., MAC translation, VLAN-ID setting, Sequence number addition, etc.) to allow proper TSN specific handling inside the sub-network. There are no requirements defined for TSN-unaware IP DetNet nodes in this document.

IP (DetNet) nodes being TSN-aware can be treated as a combination of a TSN-unaware Talker/Listener and a TSN-Relay, as shown in Figure 3. In such cases the IP (DetNet) node must provide the TSN sub-network specific Ethernet encapsulation over the link(s) towards the sub-network.

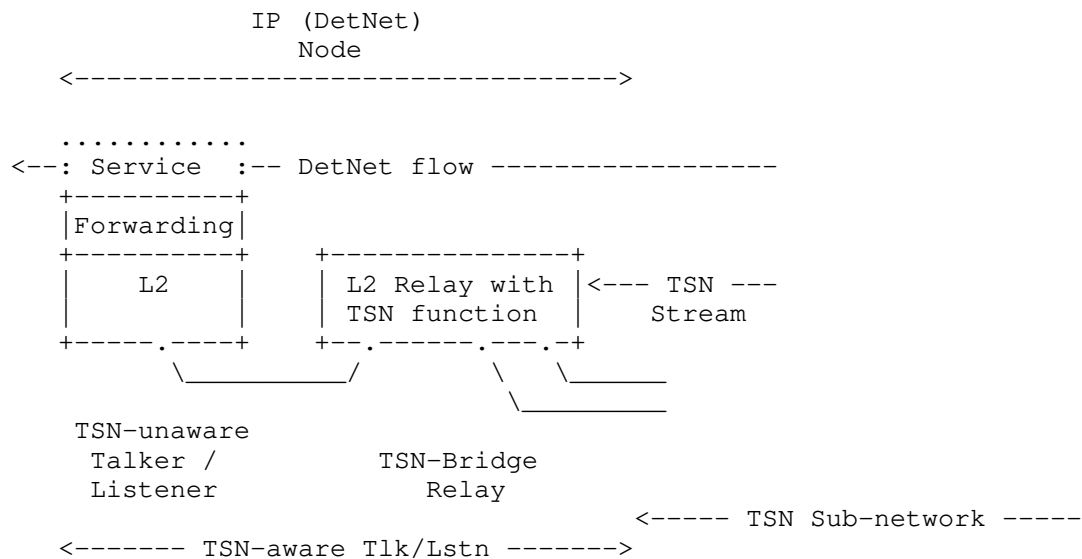


Figure 3: IP (DetNet) node with TSN functions

A TSN-aware IP (DetNet) node implementations MUST support the Stream Identification TSN component for recognizing flows.

A Stream identification component MUST be able to instantiate the following functions (1) Active Destination MAC and VLAN Stream identification function, (2) IP Stream identification function, (3) Mask-and-Match Stream identification function and (4) the related managed objects in Clause 9 of IEEE 802.1CB [IEEE8021CB] and IEEE P802.1CBdb [IEEEP8021CBdb].

A TSN-aware IP (DetNet) node implementations MUST support the Sequencing function and the Sequence encode/decode function as defined in IEEE 802.1CB [IEEE8021CB] if FRER is used inside the TSN sub-network.

The Sequence encode/decode function MUST support the Redundancy tag (R-TAG) format as per Clause 7.8 of IEEE 802.1CB [IEEE8021CB].

A TSN-aware IP (DetNet) node implementations MUST support the Stream splitting function and the Individual recovery function as defined in IEEE 802.1CB [IEEE8021CB] when the node is a replication or elimination point for FRER.

4.3. Service protection within the TSN sub-network

TSN Streams supporting DetNet flows may use Frame Replication and Elimination for Redundancy (FRER) as defined in IEEE 802.1CB [IEEE8021CB] based on the loss service requirements of the TSN Stream, which is derived from the DetNet service requirements of the DetNet mapped flow. The specific operation of FRER is not modified by the use of DetNet and follows IEEE 802.1CB [IEEE8021CB].

FRER function and the provided service recovery is available only within the TSN sub-network as the TSN Stream-ID and the TSN sequence number are not valid outside the sub-network. An IP (DetNet) node represents a L3 border and as such it terminates all related information elements encoded in the L2 frames.

4.4. Aggregation during DetNet flow to TSN Stream mapping

Implementations of this document SHALL use management and control information to map a DetNet flow to a TSN Stream. N:1 mapping (aggregating DetNet flows in a single TSN Stream) SHALL be supported. The management or control function that provisions flow mapping SHALL ensure that adequate resources are allocated and configured to provide proper service requirements of the mapped flows.

5. Management and Control Implications

[Editor's note: This section covers management/control plane related implications of creation, mapping, removal of TSN Stream IDs, their related parameters and, when needed, the configuration of FRER.]

DetNet flow and TSN Stream mapping related information are required only for TSN-aware IP (DetNet) nodes. From the Data Plane perspective there is no practical difference based on the origin of flow mapping related information (management plane or control plane).

TSN-aware IP DetNet nodes are member of both the DetNet domain and the TSN sub-network. Within the TSN sub-network the TSN-aware IP (DetNet) node has a TSN-aware Talker/Listener role, so TSN specific management and control plane functionalities must be implemented. There are many similarities in the management plane techniques used

in DetNet and TSN, but that is not the case for the control plane protocols. For example, RSVP-TE and MSRP behaves differently. Therefore management and control plane design is an important aspect of scenarios, where mapping between DetNet and TSN is required.

In order to use a TSN sub-network between DetNet nodes, DetNet specific information must be converted to TSN sub-network specific ones. DetNet flow ID and flow related parameters/requirements must be converted to a TSN Stream ID and stream related parameters/requirements. Note that, as the TSN sub-network is just a portion of the end2end DetNet path (i.e., single hop from IP perspective), some parameters (e.g., delay) may differ significantly. Other parameters (like bandwidth) also may have to be tuned due to the L2 encapsulation used within the TSN sub-network.

In some case it may be challenging to determine some TSN Stream related information. For example, on a TSN-aware IP (DetNet) node that acts as a Talker, it is quite obvious which DetNet node is the Listener of the mapped TSN stream (i.e., the IP Next-Hop). However it may be not trivial to locate the point/interface where that Listener is connected to the TSN sub-network. Such attributes may require interaction between control and management plane functions and between DetNet and TSN domains.

Mapping between DetNet flow identifiers and TSN Stream identifiers, if not provided explicitly, can be done by a TSN-aware IP (DetNet) node locally based on information provided for configuration of the TSN Stream identification functions (IP Stream identification, Mask-and-match Stream identification and active Stream identification function).

Triggering the setup/modification of a TSN Stream in the TSN sub-network is an example where management and/or control plane interactions are required between the DetNet and TSN sub-network. TSN-unaware IP (DetNet) nodes make such a triggering even more complicated as they are fully unaware of the sub-network and run independently.

Configuration of TSN specific functions (e.g., FRER) inside the TSN sub-network is a TSN domain specific decision and may not be visible in the DetNet domain.

6. Security Considerations

The security considerations of DetNet in general are discussed in [I-D.ietf-detnet-architecture] and [I-D.ietf-detnet-security]. DetNet IP data plane specific considerations are summarized in

[I-D.ietf-detnet-ip]. Encryption may provided by an underlying sub-net using MACSec [IEEE802.1AE-2018] for DetNet IP over TSN flows.

7. IANA Considerations

None.

8. Acknowledgements

The authors wish to thank Norman Finn, Lou Berger, Craig Gunther, Christophe Mangin and Jouni Korhonen for their various contributions to this work.

9. References

9.1. Normative references

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

9.2. Informative references

- [G.8275.1] International Telecommunication Union, "Precision time protocol telecom profile for phase/time synchronization with full timing support from the network", ITU-T G.8275.1/Y.1369.1 G.8275.1, June 2016, <<https://www.itu.int/rec/T-REC-G.8275.1/en>>.
- [G.8275.2] International Telecommunication Union, "Precision time protocol telecom profile for phase/time synchronization with partial timing support from the network", ITU-T G.8275.2/Y.1369.2 G.8275.2, June 2016, <<https://www.itu.int/rec/T-REC-G.8275.2/en>>.
- [I-D.ietf-detnet-architecture] Finn, N., Thubert, P., Varga, B., and J. Farkas, "Deterministic Networking Architecture", draft-ietf-detnet-architecture-13 (work in progress), May 2019.

- [I-D.ietf-detnet-flow-information-model]
Farkas, J., Varga, B., Cummings, R., Jiang, Y., and D. Fedyk, "DetNet Flow Information Model", draft-ietf-detnet-flow-information-model-05 (work in progress), September 2019.
- [I-D.ietf-detnet-ip]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: IP", draft-ietf-detnet-ip-01 (work in progress), July 2019.
- [I-D.ietf-detnet-security]
Mizrahi, T., Grossman, E., Hacker, A., Das, S., Dowdell, J., Austad, H., Stanton, K., and N. Finn, "Deterministic Networking (DetNet) Security Considerations", draft-ietf-detnet-security-05 (work in progress), August 2019.
- [IEEE1588]
IEEE, "IEEE 1588 Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems Version 2", 2008.
- [IEEE802.1AE-2018]
IEEE Standards Association, "IEEE Std 802.1AE-2018 MAC Security (MACsec)", 2018,
<<https://ieeexplore.ieee.org/document/8585421>>.
- [IEEE8021CB]
Finn, N., "Draft Standard for Local and metropolitan area networks -- Seamless Redundancy", IEEE P802.1CB /D2.1 P802.1CB, December 2015,
<<http://www.ieee802.org/1/files/private/cb-drafts/d2/802-1CB-d2-1.pdf>>.
- [IEEE8021Q]
IEEE 802.1, "Standard for Local and metropolitan area networks--Bridges and Bridged Networks (IEEE Std 802.1Q-2014)", 2014, <<http://standards.ieee.org/about/get/>>.
- [IEEEP8021CBdb]
Mangin, C., "Extended Stream identification functions", IEEE P802.1CBdb /D0.2 P802.1CBdb, August 2019,
<<http://www.ieee802.org/1/files/private/cb-drafts/d2/802-1CB-d2-1.pdf>>.

Authors' Addresses

Balazs Varga (editor)
Ericsson
Magyar Tudosok krt. 11.
Budapest 1117
Hungary

Email: balazs.a.varga@ericsson.com

Janos Farkas
Ericsson
Magyar Tudosok krt. 11.
Budapest 1117
Hungary

Email: janos.farkas@ericsson.com

Andrew G. Malis
Independent

Email: agmalis@gmail.com

Stewart Bryant
Futurewei Technologies

Email: stewart.bryant@gmail.com

DetNet
Internet-Draft
Intended status: Standards Track
Expires: April 29, 2020

B. Varga, Ed.
J. Farkas
Ericsson
L. Berger
D. Fedyk
LabN Consulting, L.L.C.
A. Malis
Independent
S. Bryant
Futurewei Technologies
J. Korhonen
October 27, 2019

DetNet Data Plane: MPLS
draft-ietf-detnet-mpls-03

Abstract

This document specifies the Deterministic Networking data plane when operating over an MPLS Packet Switched Networks.

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 April 29, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents

carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Terminology	3
2.1. Terms Used in This Document	3
2.2. Abbreviations	4
2.3. Requirements Language	5
3. DetNet MPLS Data Plane Overview	5
3.1. Layers of DetNet Data Plane	5
3.2. DetNet MPLS Data Plane Scenarios	6
4. MPLS-Based DetNet Data Plane Solution	8
4.1. DetNet Over MPLS Encapsulation Components	8
4.2. MPLS Data Plane Encapsulation	9
4.2.1. DetNet Control Word and the DetNet Sequence Number	10
4.2.2. S-Labels	11
4.2.3. F-Labels	14
4.3. OAM Indication	16
4.4. Flow Aggregation	17
4.4.1. Aggregation Via LSP Hierarchy	17
4.4.2. Aggregating DetNet Flows as a new DetNet flow	17
4.5. Service Sub-Layer Considerations	19
4.5.1. Edge Node Processing	19
4.5.2. Relay Node Processing	19
4.6. Forwarding Sub-Layer Considerations	20
4.6.1. Class of Service	20
4.6.2. Quality of Service	20
5. Management and Control Information Summary	21
5.1. Service Sub-Layer Information Summary	21
5.1.1. Service Aggregation Information Summary	22
5.2. Forwarding Sub-Layer Information Summary	23
6. Security Considerations	24
7. IANA Considerations	25
8. Acknowledgements	25
9. References	25
9.1. Normative References	25
9.2. Informative References	27
Authors' Addresses	29

1. Introduction

Deterministic Networking (DetNet) is a service that can be offered by a network to DetNet flows. DetNet provides these flows extremely low packet loss rates and assured maximum end-to-end delivery latency. General background and concepts of DetNet can be found in [I-D.ietf-detnet-architecture].

The DetNet Architecture models the DetNet related data plane functions decomposed into two sub-layers: a service sub-layer and a forwarding sub-layer. The service sub-layer is used to provide DetNet service functions such as protection and reordering. The forwarding sub-layer is used to provide forwarding assurance (low loss, assured latency, and limited reordering).

This document specifies the DetNet data plane operation and the on-wire encapsulation of DetNet flows over an MPLS-based Packet Switched Network (PSN) using the service sub-layer reference model. MPLS encapsulation already provides a solid foundation of building blocks to enable the DetNet service and forwarding sub-layer functions. MPLS encapsulated DetNet can be carried over a variety of different network technologies that can provide the DetNet required level of service. However, the specific details of how DetNet MPLS is carried over different network technologies is out of scope of this document.

MPLS encapsulated DetNet flows can carry different types of traffic. The details of the types of traffic that are carried in DetNet are also out of scope of this document. An example of IP using DetNet MPLS sub-networks can be found in [I-D.ietf-detnet-ip]. DetNet MPLS may use an associated controller and Operations, Administration, and Maintenance (OAM) functions that are defined outside of this document.

Background information common to all data planes for DetNet can be found in the DetNet Data Plane Framework [I-D.ietf-detnet-data-plane-framework].

2. Terminology

2.1. Terms Used in This Document

This document uses the terminology established in the DetNet architecture [I-D.ietf-detnet-architecture] and the the DetNet Data Plane Framework [I-D.ietf-detnet-data-plane-framework]. The reader is assumed to be familiar with these documents and any terminology defined therein.

The following terminology is introduced in this document:

F-Label	A Detnet "forwarding" label that identifies the LSP used to forward a DetNet flow across an MPLS PSN, e.g., a hop-by-hop label used between label switching routers (LSR).
S-Label	A DetNet "service" label that is used between DetNet nodes that implement also the DetNet service sub-layer functions. An S-Label is also used to identify a DetNet flow at DetNet service sub-layer.
A-Label	A special case of an S-Label, whose aggregation properties are known only at the aggregation and deaggregation end-points.
d-CW	A DetNet Control Word (d-CW) is used for sequencing information of a DetNet flow at the DetNet service sub-layer.

2.2. Abbreviations

The following abbreviations are used in this document:

CoS	Class of Service.
CW	Control Word.
DetNet	Deterministic Networking.
LSR	Label Switching Router.
MPLS	Multiprotocol Label Switching.
MPLS-TE	Multiprotocol Label Switching - Traffic Engineering.
MPLS-TP	Multiprotocol Label Switching - Transport Profile.
OAM	Operations, Administration, and Maintenance.
PE	Provider Edge.
PEF	Packet Elimination Function.
PRF	Packet Replication Function.
PREOF	Packet Replication, Elimination and Ordering Functions.
POF	Packet Ordering Function.

PSN	Packet Switched Network.
PW	PseudoWire.
QoS	Quality of Service.
S-PE	Switching Provider Edge.
T-PE	Terminating Provider Edge.
TSN	Time-Sensitive Network.

2.3. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. DetNet MPLS Data Plane Overview

3.1. Layers of DetNet Data Plane

MPLS provides a wide range of capabilities that can be utilised by DetNet. A straight forward approach utilizing MPLS for a DetNet service sub-layer is based on the existing pseudowire (PW) encapsulations and by utilizing existing MPLS Traffic Engineering encapsulations and mechanisms. Background on PWs can be found in [RFC3985] and [RFC3031]. Background on MPLS Traffic Engineering can be found in [RFC3272] and [RFC3209].

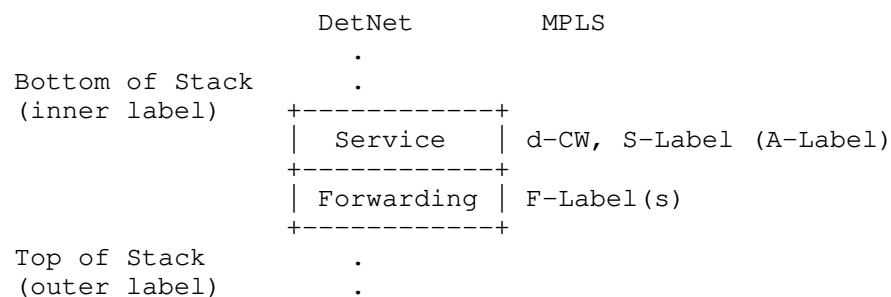


Figure 1: DetNet Adaptation to MPLS Data Plane

The DetNet MPLS data plane representation is illustrated in Figure 1. The service sub-layer includes a DetNet control word (d-CW) and a identifying service label (S-Label). The DetNet control word (d-CW) conforms to the Generic PW MPLS Control Word (PWMCW) defined in [RFC4385]. An aggregation label (A-Label) is a special case of S-Label used for aggregation.

A node operating on a DetNet flow in the Detnet service sub-layer, uses the local context associated with that S-Label, provided by a received F-Label, to determine what local DetNet operation(s) are applied to that packet. An S-Label may be taken from the platform label space [RFC3031], making it unique, enabling DetNet flow identification regardless of which input interface or LSP the packet arrives on.

The DetNet forwarding sub-layer is supported by zero or more forwarding labels (F-Labels). MPLS Traffic Engineering encapsulations and mechanisms can be utilized to provide a forwarding sub-layer that is responsible for providing resource allocation and explicit routes.

3.2. DetNet MPLS Data Plane Scenarios

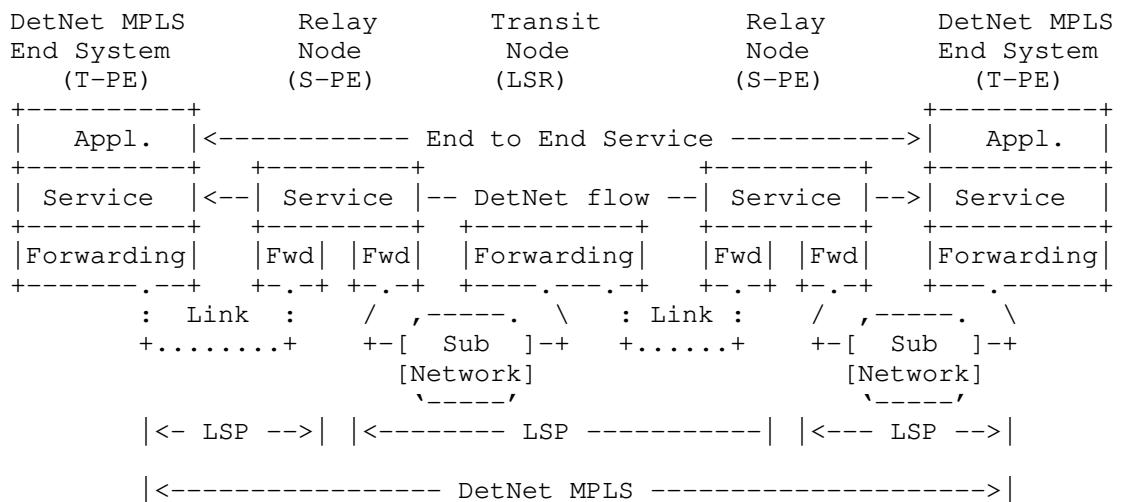


Figure 2: A DetNet MPLS Network

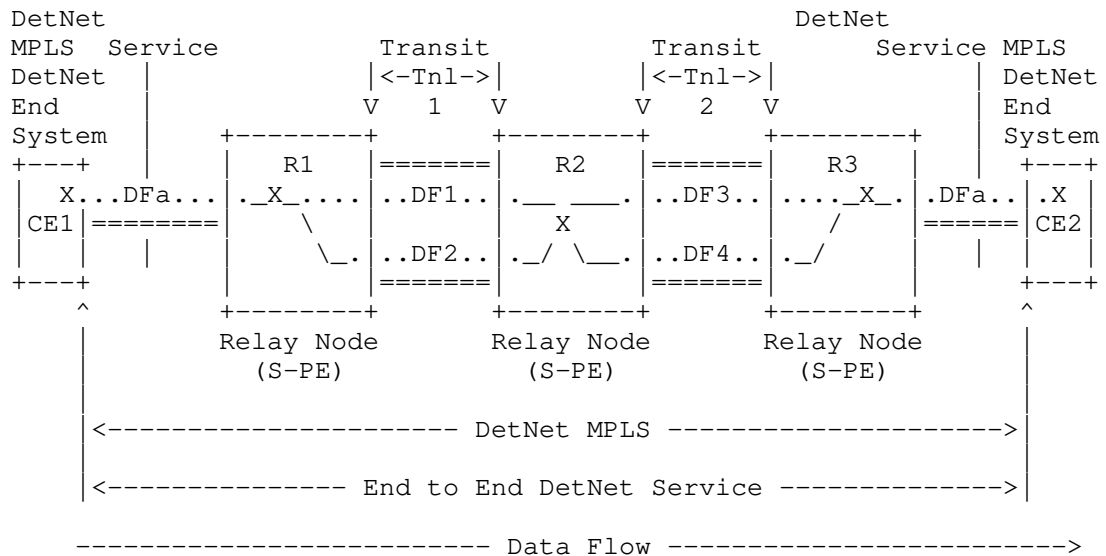
Figure 2 illustrates a hypothetical DetNet MPLS-only network composed of DetNet aware MPLS enabled end systems, operating over a DetNet aware MPLS network. In this figure, the relay nodes are PE devices that define the MPLS LSP boundaries and the transit nodes are LSRs.

DetNet end system and relay nodes understand the particular needs of DetNet flows and provide both DetNet service and forwarding sub-layer functions. In the case of MPLS, DetNet service-aware nodes add, remove and process d-CWs, S-Labels and F-labels as needed. DetNet MPLS nodes provide functionality analogous to T-PEs when they sit at the edge of an MPLS domain, and S-PEs when they are in the middle of an MPLS domain, see [RFC6073].

In a DetNet MPLS network, transit nodes may be DetNet service aware or may be DetNet unaware MPLS Label Switching Routers (LSRs). In this latter case, such LSRs would be unaware of the special requirements of the DetNet service sub-layer, but would still provide traffic engineering functions and the QoS capabilities needed to ensure that the (TE) LSPs meet the service requirements of the carried DetNet flows.

The application of DetNet using MPLS supports a number of control plane/management plane types. These types support certain MPLS data plane deployments. For example RSVP-TE, MPLS-TP, or MPLS Segment Routing (when extended to support resource allocation) are all valid MPLS deployment cases.

Figure 3 illustrates how an end-to-end MPLS-based DetNet service is provided in a more detail. In this figure, the customer end systems, CE1 and CE2, are able to send and receive MPLS encapsulated DetNet flows, and R1, R2 and R3 are relay nodes in the middle of a DetNet network. The 'X' in the end systems, and relay nodes represents potential DetNet compound flow packet replication and elimination points. In this example, service protection is supported utilizing at least two DetNet member flows and TE LSPs. For a unidirectional flow, R1 supports PRF and R3 supports PEF and POF. Note that the relay nodes may change the underlying forwarding sub-layer, for example tunneling MPLS over IEEE 802.1 TSN [I-D.ietf-detnet-mpls-over-tsn], or simply over interconnect network links.



X = Optional service protection (none, PRF, PREOF, PEF/POF)
 DFx = DetNet member flow x over a TE LSP

Figure 3: MPLS-Based Native DetNet

4. MPLS-Based DetNet Data Plane Solution

4.1. DetNet Over MPLS Encapsulation Components

To carry DetNet over MPLS the following is required:

1. A method of identifying the MPLS payload type.
2. A method of identifying the DetNet flow group to the processing element.
3. A method of distinguishing DetNet OAM packets from DetNet data packets.
4. A method of carrying the DetNet sequence number.
5. A suitable LSP to deliver the packet to the egress PE.
6. A method of carrying queuing and forwarding indication.

In this design an MPLS service label (the S-Label), similar to a pseudowire (PW) label [RFC3985], is used to identify both the DetNet flow identity and the payload MPLS payload type satisfying (1) and

(2) in the list above. OAM traffic discrimination happens through the use of the Associated Channel method described in [RFC4385]. The DetNet sequence number is carried in the DetNet Control word which carries the Data/OAM discriminator. To simplify implementation and to maximize interoperability two sequence number sizes are supported: a 16 bit sequence number and a 28 bit sequence number. The 16 bit sequence number is needed to support some types of legacy clients. The 28 bit sequence number is used in situations where it is necessary ensure that in high speed networks the sequence number space does not wrap whilst packets are in flight.

The LSP used to forward the DetNet packet may be of any type (MPLS-LDP, MPLS-TE, MPLS-TP [RFC5921], or MPLS-SR [I-D.ietf-spring-segment-routing-mpls]). The LSP (F-Label) label and/or the S-Label may be used to indicate the queue processing as well as the forwarding parameters. Note that the possible use of Penultimate Hop Popping (PHP) means that the S-Label may be the only label received at the terminating DetNet service.

4.2. MPLS Data Plane Encapsulation

Figure 4 illustrates a DetNet data plane MPLS encapsulation. The MPLS-based encapsulation of the DetNet flows is well suited for the scenarios described in [I-D.ietf-detnet-data-plane-framework]. Furthermore, an end to end DetNet service i.e., native DetNet deployment (see Section 3.2) is also possible if DetNet end systems are capable of initiating and termination MPLS encapsulated packets.

The MPLS-based DetNet data plane encapsulation consists of:

- o DetNet control word (d-CW) containing sequencing information for packet replication and duplicate elimination purposes, and the OAM indicator.
- o DetNet service Label (S-Label) that identifies a DetNet flow at the receiving DetNet service sub-layer processing node.
- o Zero or more Detnet MPLS Forwarding label(s) (F-Label) used to direct the packet along the label switched path (LSP) to the next service sub-layer processing node along the path. When Penultimate Hop Popping is in use there may be no label F-Label in the protocol stack on the final hop.
- o The necessary data-link encapsulation is then applied prior to transmission over the physical media.

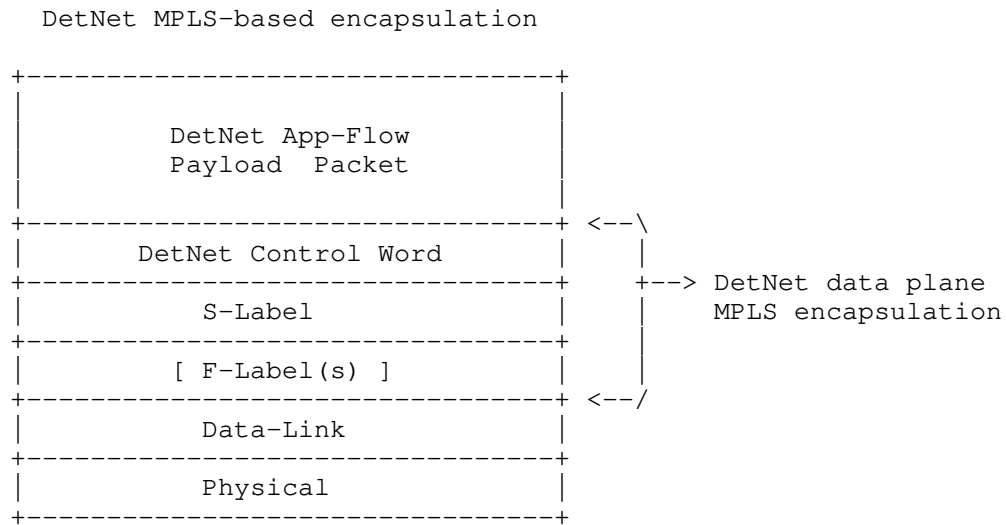


Figure 4: Encapsulation of a DetNet App-Flow in an MPLS PSN

4.2.1. DetNet Control Word and the DetNet Sequence Number

A DetNet control word (d-CW) conforms to the Generic PW MPLS Control Word (PVMCW) defined in [RFC4385]. The d-CW formatted as shown in Figure 5 MUST be present in all DetNet packets containing app-flow data.

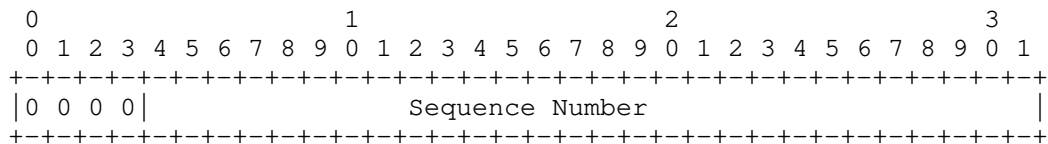


Figure 5: DetNet Control Word

(bits 0 to 3)

Per [RFC4385], MUST be set to zero (0).

Sequence Number (bits 4 to 31)

An unsigned value implementing the DetNet sequence number.

A separate sequence number space MUST be maintained by the node that adds the d-CW for each DetNet app-flow. The following sequence number field lengths MUST be supported:

0 bits

16 bits

28 bits

The sequence number length MUST be provisioned on a per app-flow basis via configuration, i.e., via the controller plane described in [I-D.ietf-detnet-data-plane-framework].

A 0 bit sequence number field length indicates that there is no DetNet sequence number used for the flow. When the length is zero, the sequence number field MUST be set to zero (0) on all packets sent for the flow.

When the sequence number field length is 16 or 28 bits for a flow, the sequence number MUST be incremented by one for each new app-flow packet sent. When the field length is 16 bits, d-CW bits 4 to 15 MUST be set to zero (0). The values carried in this field can wrap and it is important to note that zero (0) is a valid field value. For example, were the sequence number size is 16 bits, the sequence will contain: 65535, 0, 1, where zero (0) is an ordinary sequence number.

It is important to note that this document differs from [RFC4448] where a sequence number of zero (0) is used to indicate that the sequence number check algorithm is not used.

The sequence number is optionally used during receive processing as described below in Section 4.2.2.1 and Section 4.2.2.2.

4.2.2. S-Labels

App-flow identification at a DetNet service sub-layer is realized by an S-Label. MPLS-aware DetNet end systems and edge nodes, which are by definition MPLS ingress and egress nodes, MUST add and remove an app-flow specific d-CW and S-Label. Relay nodes MAY swap S-Label values when processing an app-flow.

The S-Label value MUST be provisioned per app-flow via configuration, e.g., via the controller plane described in [I-D.ietf-detnet-data-plane-framework]. Note that S-Labels provide app-flow identification at the downstream DetNet service sub-layer receiver, not the sender. As such, S-Labels MUST be allocated by the

entity that controls the service sub-layer receiving node's label space, and MAY be allocated from the platform label space [RFC3031]. Because S-Labels are local to each node rather than being a global identifier within a domain, they must be advertised to their upstream DetNet service-aware peer nodes (e.g., a DetNet MPLS End System or a DetNet Relay or Edge Node and interpreted in the context of their received F-Label.

The S-Label will normally be at the bottom of the label stack once the last F-Label is removed, immediately preceding the d-CW. To support service sub-layer level OAM, an OAM Associated Channel Header (ACH) [RFC4385] together with a Generic Associated Channel Label (GAL) [RFC5586] MAY be used in place of a d-CW.

Similarly, an Entropy Label Indicator/Entropy Label (ELI/EL) [RFC6790] MAY be carried below the S-Label in the label stack in networks where DetNet flows would otherwise received ECMP treatment. When ELs are used, the same EL value SHOULD be used for all of the packets sent using a specific S-Label to force the flow to follow the same path. However, as outlines in [I-D.ietf-detnet-data-plane-framework] the use of ECMP for DetNet flows is NOT RECOMMENDED. ECMP MAY be used for non-DetNet flows within a DetNet domain.

When receiving a DetNet MPLS flow, an implementation MUST identify the app-flow associated with the incoming packet based on the S-Label. When a node is using platform labels for S-Labels, no additional information is needed as the S-label uniquely identifies the app-flow. In the case where platform labels are not used, zero or more F-Labels and optionally, the incoming interface, proceeding the S-Label MUST be used together with the S-Label to uniquely identify the app-flows associated with a received packet. The incoming interface MAY also be used to together with any present F-Label(s) and the S-Label to uniquely identify an incoming app-flows, for example, to in the case where PHP is used. Note that choice to use platform label space for S-Label or S-Label plus one or more F-Labels to identify app flows is a local implementation choice, with one caveat. When one or more F-labels, or incoming interface, is needed together with an S-Label to uniquely identify, the controller plane MUST ensure that incoming DetNet MPLS packets arrive with the needed information (F-label(s) and/or incoming interface); the details of such are outside the scope of this document.

The use of platform labels for S-Labels matches other pseudowire encapsulations for consistency but there is no hard requirement in this regard.

4.2.2.1. Packet Elimination Function Processing

Implementations MAY support the Packet Elimination Function (PEF) for received DetNet MPLS flows. When supported, use of the PEF for a particular app-flow MUST be provisioned via configuration, e.g., via the controller plane described in [I-D.ietf-detnet-data-plane-framework].

After an app-flow is identified for a received DetNet MPLS packet, as described above, an implementation MUST check if PEF is configured for that app-flow. When configured, the implementation MUST track the sequence number contained in received d-CWs and MUST ensure that duplicate (replicated) instances of a particular sequence number are discarded. The specific mechanisms used for an implementation to identify which received packets are duplicates and which are new is an implementation choice. Note that per Section 4.2.1 the sequence number field length may be 16 or 28 bits, and the field value can wrap.

Note that an implementation MAY wish to constrain the maximum number sequence numbers that are tracked, on platform-wide or per flow basis. Some implementations MAY support the provisioning of the maximum number sequence numbers that are tracked number on either a platform-wide or per flow basis.

4.2.2.2. Packet Ordering Function Processing

A function that is related to in-order delivery is the Packet Ordering Function (POF). Implementations MAY support POF. When supported, use of the POF for a particular app-flow MUST be provisioned via configuration, e.g., via the controller plane described by [I-D.ietf-detnet-data-plane-framework]. Implementations MAY required that PEF and POF be used in combination. There is no requirement related to the order of execution of the Packet Elimination and Ordering Functions in an implementation.

After an app-flow is identified for a received DetNet MPLS packet, as described above, an implementation MUST check if POF is configured for that app-flow. When configured, the implementation MUST track the sequence number contained in received d-CWs and MUST ensure that packets are processed in the order indicated in the received d-CW sequence number field, which may not be in the order the packets are received. As defined in Section 4.2.1 the sequence number field length may be 16 or 28 bits, is incremented by one (1) for each new app-flow packet sent, and the field value can wrap. The specific mechanisms used for an implementation to identify the order of received packets is an implementation choice.

Note that an implementation MAY wish to constrain the maximum number of out of order packets that can be processed, on platform-wide or per flow basis. Some implementations MAY support the provisioning of this number on either a platform-wide or per flow basis. The number of out of order packets that can be processed also impacts the latency of a flow.

4.2.3. F-Labels

F-Labels are supported the DetNet forwarding sub-layer. F-Labels are used to provide LSP-based connectivity between DetNet service sub-layer processing nodes.

4.2.3.1. Service Sub-Layer and Packet Replication Function Processing

DetNet MPLS end systems, edge nodes and relay nodes may operate at the DetNet service sub-layer with understand of app-flows and their requirements. As mentioned earlier, when operating at this layer such nodes can push, pop or swap (pop then push) S-Labels. In all cases, the F-Labels used for the app-flow are always replaced and the following procedures apply.

When sending a DetNet flow, zero or more F-Labels MAY be pushed on top of an S-Label by the node pushing an S-Label. The F-Labels to be pushed when sending a particular app-flow MUST be provisioned per app-flow via configuration, e.g., via the controller plane discussed in [I-D.ietf-detnet-data-plane-framework]. F-Labels can also provide context for an S-Label. To allow for the omission of F-Labels, an implementation SHOULD also allow an outgoing interface to be used.

The Packet Replication Function (PRF) function MAY be supported by an implementation for outgoing DetNet flows. When replication is supported, the same app-flow data will be sent over multiple outgoing forwarding sub-layer LSPs. To support PRF an implementation MUST support the setting of different sets of F-Labels. To allow for the omission of F-Labels, an implementation SHOULD also allow multiple outgoing interfaces to be provisioned. PRF MUST NOT be used with app-flows configured with a d-CW sequence number field length of 0 bits.

When a single set of F-Labels is provisioned for a particular outgoing app-flow, that set of F-labels MUST be pushed after the S-Label is pushed. The outgoing packet is then forwarded as described below in Section 4.2.3.2. When a single outgoing interface is provisioned, the outgoing packet is then forwarded as described below in Section 4.2.3.2.

When multiple sets of F-Labels or interfaces are provisioned for a particular outgoing app-flow, a copy of the outgoing packet, including the pushed S-Label, MUST be made per F-label set and outgoing interface. Each set of provisioned F-Labels are then pushed onto a copy of the packet. Each copy is then forwarded as described below in Section 4.2.3.2.

As described in the previous section, when receiving a DetNet MPLS flow, an implementation identifies the app-flow associated with the incoming packet based on the S-Label. When a node is using platform labels for S-Labels, any F-Labels can be popped and the S-label uniquely identifies the app-flow. In the case where platform labels are not used, F-Label(s) and, optionally, the incoming interface MUST also be provisioned for incoming app-flows. The provisioned information MUST then be used to identify incoming app-flows based on the combination of S-Label and F-Label(s) or incoming interface.

4.2.3.2. Common F-Label Processing

All DetNet aware MPLS nodes process F-Labels as needed to meet the service requirements of the DetNet flow or flows carried in the LSPs represented by the F-Labels. This includes normal push, pop and swap operations. Such processing is essentially the same type of processing provided for TE LSPs, although the specific service parameters, or traffic specification, can differ. When the DetNet service parameters of the app-flow or flows carried in an LSP represented by an F-Label can be met by an existing TE mechanism, the forwarding sub-layer processing node MAY be a DetNet unaware, i.e., standard, MPLS LSR. Such TE LSPs may provide LSP forwarding service as defined in, but not limited to, [RFC3209], [RFC3270], [RFC3272], [RFC3473], [RFC4875], [RFC5440], and [RFC8306].

More specifically, as mentioned above, the DetNet forwarding sub-layer provides explicit routes and allocated resources, and F-Labels are used to map to each. Explicit routes are supported based on the topmost (outermost) F-Label that is pushed or swapped and the LSP that corresponds to this label. This topmost (outgoing) label MUST be associated with a provisioned outgoing interface and, for non-point-to-point outgoing interfaces, a next hop LSR. Note that this information MUST be provisioned via configuration or the controller plane. In the previously mentioned special case where there are no added F-labels and the outgoing interface is not a point-to-point interface, the outgoing interface MUST also be associated with a next hop LSR.

Resources may be allocated in a hierarchical fashion per LSP that is represented by each F-Label. Each LSP MAY be provisioned with a service parameters that dictates the specific traffic treatment to be

received by the traffic carried over that LSP. Implementations of this document MUST ensure that traffic carried over each LSP represented by one or more F-Labels receives the traffic treatment provisioned for that LSP. Typical mechanisms used to provide different treatment to different flows includes the allocation of system resources (such as queues and buffers) and provisioning or related parameters (such as shaping, and policing). Support can also be provided via an underlying network technology such IEEE802.1 TSN [I-D.ietf-detnet-mpls-over-tsn]. The specific mechanisms used by a DetNet node to ensure DetNet service delivery requirements are met for supported DetNet flows is outside the scope of this document.

Packets that are marked in a way that do not correspond to allocated resources, e.g., lack a provisioned F-Label, can disrupt the QoS offered to properly reserved DetNet flows by using resources allocated to the reserved flows. Therefore, the network nodes of a DetNet network:

- o MUST defend the DetNet QoS by discarding or remarking (to an allocated DetNet flow or non-competing non-DetNet flow) packets received that are not associated with a completed resource allocation.
- o MUST NOT use a DetNet allocated resource, e.g. a queue or shaper reserved for DetNet flows, for any packet that does match the corresponding DetNet flow.
- o MUST ensure a QoS flow does not exceed its allocated resources or provisioned service level,
- o MUST ensure a CoS flow or service class does not impact the service delivered to other flows. This requirement is similar to requirement for MPLS LSRs to that CoS LSPs do not impact the resources allocated to TE LSPs, e.g., via [RFC3473].

Subsequent sections provide additional considerations related to CoS (Section 4.6.1), QoS (Section 4.6.2) and aggregation (Section 4.4).

4.3. OAM Indication

OAM follows the procedures set out in [RFC5085] with the restriction that only Virtual Circuit Connectivity Verification (VCCV) type 1 is supported.

As shown in Figure 3 of [RFC5085] when the first nibble of the d-CW is 0x0 the payload following the d-CW is normal user data. However, when the first nibble of the d-CW is 0x1, the payload that follows

the d-DW is an OAM payload with the OAM type indicated by the value in the d-CW Channel Type field.

The reader is referred to [RFC5085] for a more detailed description of the Associated Channel mechanism, and to the DetNet work on OAM for more information DetNet OAM.

4.4. Flow Aggregation

The ability to aggregate individual flows, and their associated resource control, into a larger aggregate is an important technique for improving scaling of control in the data, management and control planes. The DetNet data plane allows for the aggregation of DetNet flows, to improved scaling. There are two methods of supporting flow aggregation covered in this section.

The resource control and management aspects of aggregation (including the configuration of queuing, shaping, and policing) are the responsibility of the DetNet controller plane and is out of scope of this documents. It is also the responsibility of the controller plane to ensure that consistent aggregation methods are used.

4.4.1. Aggregation Via LSP Hierarchy

DetNet flows forwarded via MPLS can leverage MPLS-TE's existing support for hierarchical LSPs (H-LSPs), see [RFC4206]. H-LSPs are typically used to aggregate control and resources, they may also be used to provide OAM or protection for the aggregated LSPs. Arbitrary levels of aggregation naturally falls out of the definition for hierarchy and the MPLS label stack [RFC3032]. DetNet nodes which support aggregation (LSP hierarchy) map one or more LSPs (labels) into and from an H-LSP. Both carried LSPs and H-LSPs may or may not use the TC field, i.e., L-LSPs or E-LSPs. Such nodes will need to ensure that individual LSPs and H-LSPs receive the traffic treatment required to ensure the required DetNet service is preserved.

Additional details of the traffic control capabilities needed at a DetNet-aware node may be covered in the new service definitions mentioned above or in separate future documents. Controller plane mechanisms will also need to ensure that the service required on the aggregate flow are provided, which may include the discarding or remarking mentioned in the previous sections.

4.4.2. Aggregating DetNet Flows as a new DetNet flow

An aggregate can be built by layering DetNet flows, including both their S-Label and, when present, F-Labels as shown below:

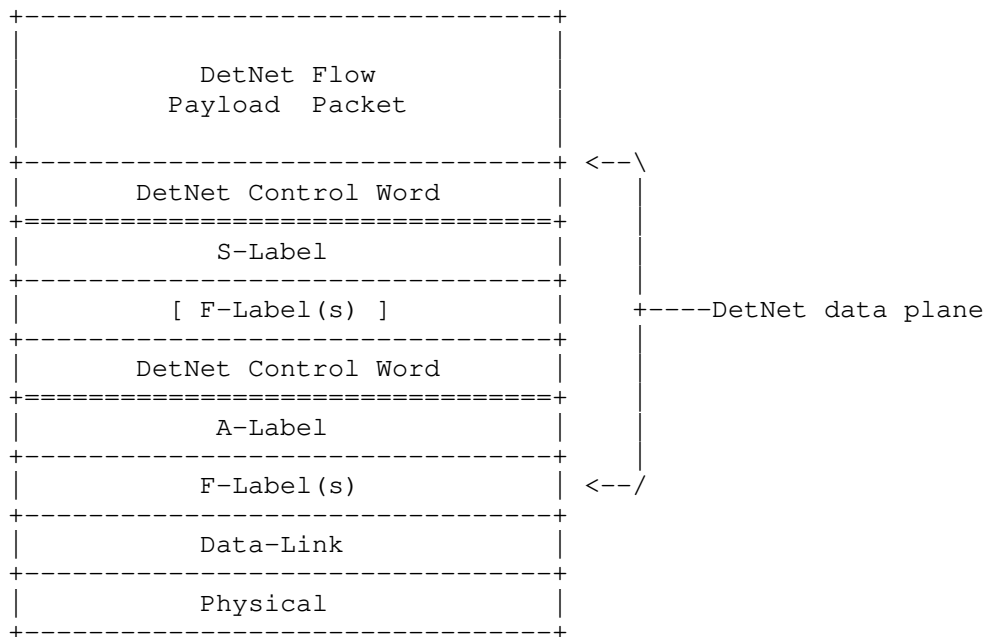


Figure 6: DetNet Aggregation as a new DetNet Flow

Both the aggregation label, which is referred to as an A-Label, and the individual flow's S-Label have their MPLS S bit set indicating bottom of stack, and the d-CW allows the PREOF to work. An A-Label is a special case of an S-Label, whose properties are known only at the aggregation and deaggregation end-points.

It is a property of the A-Label that what follows is a d-CW followed by an MPLS label stack. A relay node processing the A-Label would not know the underlying payload type, and the A-Label would be processed as a normal S-Label. This would only be known to a node that was a peer of the node imposing the S-Label. However there is no real need for it to know the payload type during aggregation processing.

As in the previous section, nodes supporting this type of aggregation will need to ensure that individual and aggregated flows receive the traffic treatment required to ensure the required DetNet service is preserved. Also, it is the controller plane's responsibility to ensure that the service required on the aggregate flow are properly provisioned.

4.5. Service Sub-Layer Considerations

The edge and relay node internal procedures related to PREOF are implementation specific. The order of a packet elimination or replication is out of scope in this specification.

It is important that the DetNet layer is configured such that a DetNet node never receives its own replicated packets. If it were to receive such packets the replication function would make the loop more destructive of bandwidth than a conventional unicast loop. Ultimately the TTL in the S-Label will cause the packet to die during a transient loop, but given the sensitivity of applications to packet latency the impact on the DetNet application would be severe. To avoid the problem of a transient forwarding loop, changes to an LSP supporting DetNet MUST be loop-free.

4.5.1. Edge Node Processing

An edge node is responsible for matching ingress packets to the service they require and encapsulating them accordingly. An edge node may participate in the packet replication and duplicate packet elimination.

The DetNet-aware forwarder selects the egress DetNet member flow segment based on the flow identification. The mapping of ingress DetNet member flow segment to egress DetNet member flow segment may be statically or dynamically configured. Additionally the DetNet-aware forwarder does duplicate frame elimination based on the flow identification and the sequence number combination. The packet replication is also done within the DetNet-aware forwarder. During elimination and the replication process the sequence number of the DetNet member flow MUST be preserved and copied to the egress DetNet member flow.

The internal design of a relay node is out of scope of this document. However the reader's attention is drawn to the need to make any PREOF state available to the packet processor(s) dealing with packets to which the PREOF functions must be applied, and to maintain that state is such as way that it is available to the packet processor operation on the next packet in the DetNet flow (which may be a duplicate, a late packet, or the next packet in sequence).

4.5.2. Relay Node Processing

A DetNet Relay node operates in the DetNet forwarding sub-layer . For DetNet using MPLS this processing is performed on the F-Label. This processing is done within an extended forwarder function. Whether an ingress DetNet member flow receives DetNet specific

processing depends on how the forwarding is programmed. Some relay nodes may be DetNet service aware, while others may be unmodified LSRs that only understand how to switch MPLS-TE LSPs.

It is also possible to treat the relay node as a transit node, see Section 4.4. Again, this is entirely up to how the forwarding has been programmed.

4.6. Forwarding Sub-Layer Considerations

4.6.1. Class of Service

Class and quality of service, i.e., CoS and QoS, are terms that are often used interchangeably and confused with each other. In the context of DetNet, CoS is used to refer to mechanisms that provide traffic forwarding treatment based on aggregate group basis and QoS is used to refer to mechanisms that provide traffic forwarding treatment based on a specific DetNet flow basis. Examples of existing network level CoS mechanisms include DiffServ which is enabled by IP header differentiated services code point (DSCP) field [RFC2474] and MPLS label traffic class field [RFC5462], and at Layer-2, by IEEE 802.1p priority code point (PCP).

CoS for DetNet flows carried in PWs and MPLS is provided using the existing MPLS Differentiated Services (DiffServ) architecture [RFC3270]. Both E-LSP and L-LSP MPLS DiffServ modes MAY be used to support DetNet flows. The Traffic Class field (formerly the EXP field) of an MPLS label follows the definition of [RFC5462] and [RFC3270]. The Uniform, Pipe, and Short Pipe DiffServ tunneling and TTL processing models are described in [RFC3270] and [RFC3443] and MAY be used for MPLS LSPs supporting DetNet flows. MPLS ECN MAY also be used as defined in ECN [RFC5129] and updated by [RFC5462].

4.6.2. Quality of Service

In addition to explicit routes, and packet replication and elimination, described in Section 4 above, DetNet provides zero congestion loss and bounded latency and jitter. As described in [I-D.ietf-detnet-architecture], there are different mechanisms that maybe used separately or in combination to deliver a zero congestion loss service. This includes Quality of Service (QoS) mechanisms at the MPLS layer, that may be combined with the mechanisms defined by the underlying network layer such as 802.1TSN.

Quality of Service (QoS) mechanisms for flow specific traffic treatment typically includes a guarantee/agreement for the service, and allocation of resources to support the service. Example QoS mechanisms include discrete resource allocation, admission control,

flow identification and isolation, and sometimes path control, traffic protection, shaping, policing and remarking. Example protocols that support QoS control include Resource ReSerVation Protocol (RSVP) [RFC2205] (RSVP) and RSVP-TE [RFC3209] and [RFC3473]. The existing MPLS mechanisms defined to support CoS [RFC3270] can also be used to reserve resources for specific traffic classes.

A baseline set of QoS capabilities for DetNet flows carried in PWs and MPLS can be provided by MPLS with Traffic Engineering (MPLS-TE) [RFC3209] and [RFC3473]. TE LSPs can also support explicit routes (path pinning). Current service definitions for packet TE LSPs can be found in "Specification of the Controlled Load Quality of Service", [RFC2211], "Specification of Guaranteed Quality of Service", [RFC2212], and "Ethernet Traffic Parameters", [RFC6003]. Additional service definitions are expected in future documents to support the full range of DetNet services. In all cases, the existing label-based marking mechanisms defined for TE-LSPs and even E-LSPs are used to support the identification of flows requiring DetNet QoS.

5. Management and Control Information Summary

The specific information needed for the processing of each DetNet service depends on the DetNet node type and the functions being provided on the node. This section summarizes based on the DetNet sub-layer and if the DetNet traffic is being sent or received. All DetNet node types are DetNet forwarding sub-layer aware, while all but transit nodes are service sub-layer aware. This is shown in Figure 2.

Much like other MPLS labels, there are a number of alternatives available for DetNet S-Label and F-Label advertisement to an upstream peer node. These include distributed signaling protocols such as RSVP-TE, centralized label distribution via a controller that manages both the sender and the receiver using NETCONF/YANG, BGP, PCEP, etc., and hybrid combinations of the two. The details of the controller plane solution required for the label distribution and the management of the label number space are out of scope of this document. There are particular DetNet considerations and requirements that are discussed in [I-D.ietf-detnet-data-plane-framework].

5.1. Service Sub-Layer Information Summary

The following summarizes the information that is needed on service sub-layer aware nodes that send DetNet MPLS traffic, on a per service basis:

- o App-Flow identification information, e.g., an incoming service on a relay node or IP information as defined in [I-D.ietf-detnet-ip-over-mpls].
- o The sequence number length to be used for the service. Valid values included 0, 16 and 28 bits. 0 bits cannot be used when PRF is configured for the service.
- o The S-Label for the service.
- o If PRF is to be provided for the service.
- o The forwarding sub-layer information associated with the output of the service sub-layer. Note that when the PRF function is provisioned, this information is per DetNet member flow. Logically the forwarding sub-layer information is a pointer to further details of transmission of Detnet flows at the forwarding sub-layer.

The following summarizes the information that is needed on service sub-layer aware nodes that receives DetNet MPLS traffic, on a per service basis:

- o The forwarding sub-layer information associated with the input of the service sub-layer. Note that when the PEF function is provisioned, this information is per DetNet member flow. Logically the forwarding sub-layer information is a pointer to further details of the reception of Detnet flows at the forwarding sub-layer or A-Label.
- o The S-Label for the received service.
- o If PEF or POF is to be provided for the service.
- o The sequence number length to be used for the service. Valid values included 0, 16 and 28 bits. 0 bits cannot be used when PEF or POF are configured for the service.

5.1.1. Service Aggregation Information Summary

Nodes performing aggregation using A-Labels, per Section 4.4.2, require the additional information summarized in this section.

The following summarizes the information that is needed on a node that sends aggregated flows using A-Labels:

- o The S-Labels or F-Labels that are to be carried over each aggregated service.
- o The A-Label associated with each aggregated service.
- o The other S-Label information summarized above.

On the receiving node, the A-Label provides the forwarding context of an incoming interface or an F-Label and is used in subsequent service or forwarding sub-layer receive processing, as appropriated. The related addition configuration that may be provided discussed elsewhere in this section.

5.2. Forwarding Sub-Layer Information Summary

The following summarizes the information that is needed on forwarding sub-layer aware nodes that send DetNet MPLS traffic, on a per forwarding sub-layer flow basis:

- o The outgoing F-Label stack to be pushed. The stack may include H-LSP labels.
- o The traffic parameters associated with a specific label in the stack. Note that there may be multiple sets of traffic parameters associated with specific labels in the stack, e.g., when H-LSPs are used.
- o Outgoing interface and, for unicast traffic, the next hop information.
- o Sub-network specific parameters on a technology specific basis. For example, see [I-D.ietf-detnet-mpls-over-tsn].

The following summarizes the information that is needed on forwarding sub-layer aware nodes that receive DetNet MPLS traffic, on a per forwarding sub-layer flow basis:

- o The incoming interface. For some implementations and deployment scenarios, this information may not be needed.
- o The incoming F-Label stack to be popped. The stack may include H-LSP labels.
- o How the incoming forwarding sub-layer flow is to be handled, i.e., forwarded as a transit node, or provided to the service sub-layer.

It is the responsibility of the DetNet controller plane to properly provision both flow identification information and the flow specific

resources needed to provided the traffic treatment needed to meet each flow's service requirements. This applies for aggregated and individual flows.

6. Security Considerations

Security considerations for DetNet are described in detail in [I-D.ietf-detnet-security]. General security considerations are described in [I-D.ietf-detnet-architecture]. This section considers exclusively security considerations which are specific to the DetNet MPLS data plane.

Security aspects which are unique to DetNet are those whose aim is to provide the specific quality of service aspects of DetNet, which are primarily to deliver data flows with extremely low packet loss rates and bounded end-to-end delivery latency.

The primary considerations for the data plane is to maintain integrity of data and delivery of the associated DetNet service traversing the DetNet network. Application flows can be protected through whatever means is provided by the underlying technology. For example, encryption may be used, such as that provided by IPSec [RFC4301] for IP flows and/or by an underlying sub-net using MACSec [IEEE802.1AE-2018] for IP over Ethernet (Layer-2) flows.

From a data plane perspective this document does not add or modify any header information.

At the management and control level DetNet flows are identified on a per-flow basis, which may provide controller plane attackers with additional information about the data flows (when compared to controller planes that do not include per-flow identification). This is an inherent property of DetNet which has security implications that should be considered when determining if DetNet is a suitable technology for any given use case.

To provide uninterrupted availability of the DetNet service, provisions can be made against DOS attacks and delay attacks. To protect against DOS attacks, excess traffic due to malicious or malfunctioning devices can be prevented or mitigated, for example through the use of existing mechanism such as policing and shaping applied at the input of a DetNet domain. To prevent DetNet packets from being delayed by an entity external to a DetNet domain, DetNet technology definition can allow for the mitigation of Man-In-The-Middle attacks, for example through use of authentication and authorization of devices within the DetNet domain.

7. IANA Considerations

This document makes no IANA requests.

8. Acknowledgements

The authors wish to thank Pat Thaler, Norman Finn, Loa Anderson, David Black, Rodney Cummings, Ethan Grossman, Tal Mizrahi, David Mozes, Craig Gunther, George Swallow, Yuanlong Jiang and Carlos J. Bernardos for their various contributions to this work.

9. References

9.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC2211] Wroclawski, J., "Specification of the Controlled-Load Network Element Service", RFC 2211, DOI 10.17487/RFC2211, September 1997, <<https://www.rfc-editor.org/info/rfc2211>>.
- [RFC2212] Shenker, S., Partridge, C., and R. Guerin, "Specification of Guaranteed Quality of Service", RFC 2212, DOI 10.17487/RFC2212, September 1997, <<https://www.rfc-editor.org/info/rfc2212>>.
- [RFC3031] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, DOI 10.17487/RFC3031, January 2001, <<https://www.rfc-editor.org/info/rfc3031>>.
- [RFC3032] Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y., Farinacci, D., Li, T., and A. Conta, "MPLS Label Stack Encoding", RFC 3032, DOI 10.17487/RFC3032, January 2001, <<https://www.rfc-editor.org/info/rfc3032>>.
- [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001, <<https://www.rfc-editor.org/info/rfc3209>>.

- [RFC3270] Le Faucheur, F., Wu, L., Davie, B., Davari, S., Vaananen, P., Krishnan, R., Cheval, P., and J. Heinanen, "Multi-Protocol Label Switching (MPLS) Support of Differentiated Services", RFC 3270, DOI 10.17487/RFC3270, May 2002, <<https://www.rfc-editor.org/info/rfc3270>>.
- [RFC3443] Agarwal, P. and B. Akyol, "Time To Live (TTL) Processing in Multi-Protocol Label Switching (MPLS) Networks", RFC 3443, DOI 10.17487/RFC3443, January 2003, <<https://www.rfc-editor.org/info/rfc3443>>.
- [RFC3473] Berger, L., Ed., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions", RFC 3473, DOI 10.17487/RFC3473, January 2003, <<https://www.rfc-editor.org/info/rfc3473>>.
- [RFC4206] Kompella, K. and Y. Rekhter, "Label Switched Paths (LSP) Hierarchy with Generalized Multi-Protocol Label Switching (GMPLS) Traffic Engineering (TE)", RFC 4206, DOI 10.17487/RFC4206, October 2005, <<https://www.rfc-editor.org/info/rfc4206>>.
- [RFC4385] Bryant, S., Swallow, G., Martini, L., and D. McPherson, "Pseudowire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN", RFC 4385, DOI 10.17487/RFC4385, February 2006, <<https://www.rfc-editor.org/info/rfc4385>>.
- [RFC5085] Nadeau, T., Ed. and C. Pignataro, Ed., "Pseudowire Virtual Circuit Connectivity Verification (VCCV): A Control Channel for Pseudowires", RFC 5085, DOI 10.17487/RFC5085, December 2007, <<https://www.rfc-editor.org/info/rfc5085>>.
- [RFC5129] Davie, B., Briscoe, B., and J. Tay, "Explicit Congestion Marking in MPLS", RFC 5129, DOI 10.17487/RFC5129, January 2008, <<https://www.rfc-editor.org/info/rfc5129>>.
- [RFC5462] Andersson, L. and R. Asati, "Multiprotocol Label Switching (MPLS) Label Stack Entry: "EXP" Field Renamed to "Traffic Class" Field", RFC 5462, DOI 10.17487/RFC5462, February 2009, <<https://www.rfc-editor.org/info/rfc5462>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

9.2. Informative References

- [I-D.ietf-detnet-architecture]
Finn, N., Thubert, P., Varga, B., and J. Farkas,
"Deterministic Networking Architecture", draft-ietf-
detnet-architecture-13 (work in progress), May 2019.
- [I-D.ietf-detnet-data-plane-framework]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A.,
Bryant, S., and J. Korhonen, "DetNet Data Plane
Framework", draft-ietf-detnet-data-plane-framework-02
(work in progress), September 2019.
- [I-D.ietf-detnet-ip]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A.,
Bryant, S., and J. Korhonen, "DetNet Data Plane: IP",
draft-ietf-detnet-ip-01 (work in progress), July 2019.
- [I-D.ietf-detnet-ip-over-mpls]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A.,
Bryant, S., and J. Korhonen, "DetNet Data Plane: IP over
MPLS", draft-ietf-detnet-ip-over-mpls-01 (work in
progress), July 2019.
- [I-D.ietf-detnet-mpls-over-tsn]
Varga, B., Farkas, J., Malis, A., Bryant, S., and J.
Korhonen, "DetNet Data Plane: MPLS over IEEE 802.1 Time
Sensitive Networking (TSN)", draft-ietf-detnet-mpls-over-
tsn-00 (work in progress), May 2019.
- [I-D.ietf-detnet-security]
Mizrahi, T., Grossman, E., Hacker, A., Das, S., Dowdell,
J., Austad, H., Stanton, K., and N. Finn, "Deterministic
Networking (DetNet) Security Considerations", draft-ietf-
detnet-security-05 (work in progress), August 2019.
- [I-D.ietf-spring-segment-routing-mpls]
Bashandy, A., Filsfils, C., Previdi, S., Decraene, B.,
Litkowski, S., and R. Shakir, "Segment Routing with MPLS
data plane", draft-ietf-spring-segment-routing-mpls-22
(work in progress), May 2019.
- [IEEE802.1AE-2018]
IEEE Standards Association, "IEEE Std 802.1AE-2018 MAC
Security (MACsec)", 2018,
<<https://ieeexplore.ieee.org/document/8585421>>.

- [RFC2205] Braden, R., Ed., Zhang, L., Berson, S., Herzog, S., and S. Jamin, "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", RFC 2205, DOI 10.17487/RFC2205, September 1997, <<https://www.rfc-editor.org/info/rfc2205>>.
- [RFC2474] Nichols, K., Blake, S., Baker, F., and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, DOI 10.17487/RFC2474, December 1998, <<https://www.rfc-editor.org/info/rfc2474>>.
- [RFC3272] Awduche, D., Chiu, A., Elwalid, A., Widjaja, I., and X. Xiao, "Overview and Principles of Internet Traffic Engineering", RFC 3272, DOI 10.17487/RFC3272, May 2002, <<https://www.rfc-editor.org/info/rfc3272>>.
- [RFC3985] Bryant, S., Ed. and P. Pate, Ed., "Pseudo Wire Emulation Edge-to-Edge (PWE3) Architecture", RFC 3985, DOI 10.17487/RFC3985, March 2005, <<https://www.rfc-editor.org/info/rfc3985>>.
- [RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <<https://www.rfc-editor.org/info/rfc4301>>.
- [RFC4448] Martini, L., Ed., Rosen, E., El-Aawar, N., and G. Heron, "Encapsulation Methods for Transport of Ethernet over MPLS Networks", RFC 4448, DOI 10.17487/RFC4448, April 2006, <<https://www.rfc-editor.org/info/rfc4448>>.
- [RFC4875] Aggarwal, R., Ed., Papadimitriou, D., Ed., and S. Yasukawa, Ed., "Extensions to Resource Reservation Protocol - Traffic Engineering (RSVP-TE) for Point-to-Multipoint TE Label Switched Paths (LSPs)", RFC 4875, DOI 10.17487/RFC4875, May 2007, <<https://www.rfc-editor.org/info/rfc4875>>.
- [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <<https://www.rfc-editor.org/info/rfc5440>>.
- [RFC5586] Bocci, M., Ed., Vigoureux, M., Ed., and S. Bryant, Ed., "MPLS Generic Associated Channel", RFC 5586, DOI 10.17487/RFC5586, June 2009, <<https://www.rfc-editor.org/info/rfc5586>>.

- [RFC5921] Bocci, M., Ed., Bryant, S., Ed., Frost, D., Ed., Levrau, L., and L. Berger, "A Framework for MPLS in Transport Networks", RFC 5921, DOI 10.17487/RFC5921, July 2010, <<https://www.rfc-editor.org/info/rfc5921>>.
- [RFC6003] Papadimitriou, D., "Ethernet Traffic Parameters", RFC 6003, DOI 10.17487/RFC6003, October 2010, <<https://www.rfc-editor.org/info/rfc6003>>.
- [RFC6073] Martini, L., Metz, C., Nadeau, T., Bocci, M., and M. Aissaoui, "Segmented Pseudowire", RFC 6073, DOI 10.17487/RFC6073, January 2011, <<https://www.rfc-editor.org/info/rfc6073>>.
- [RFC6790] Kompella, K., Drake, J., Amante, S., Henderickx, W., and L. Yong, "The Use of Entropy Labels in MPLS Forwarding", RFC 6790, DOI 10.17487/RFC6790, November 2012, <<https://www.rfc-editor.org/info/rfc6790>>.
- [RFC8306] Zhao, Q., Dhody, D., Ed., Palleti, R., and D. King, "Extensions to the Path Computation Element Communication Protocol (PCEP) for Point-to-Multipoint Traffic Engineering Label Switched Paths", RFC 8306, DOI 10.17487/RFC8306, November 2017, <<https://www.rfc-editor.org/info/rfc8306>>.

Authors' Addresses

Balazs Varga (editor)
Ericsson
Magyar Tudosok krt. 11.
Budapest 1117
Hungary

Email: balazs.a.varga@ericsson.com

Janos Farkas
Ericsson
Magyar Tudosok krt. 11.
Budapest 1117
Hungary

Email: janos.farkas@ericsson.com

Lou Berger
LabN Consulting, L.L.C.

Email: lberger@labn.net

Don Fedyk
LabN Consulting, L.L.C.

Email: dfedyk@labn.net

Andrew G. Malis
Independent

Email: agmalis@gmail.com

Stewart Bryant
Futurewei Technologies

Email: stewart.bryant@gmail.com

Jouni Korhonen

Email: jouni.nospam@gmail.com

DetNet
Internet-Draft
Intended status: Standards Track
Expires: April 29, 2020

B. Varga, Ed.
J. Farkas
Ericsson
A. Malis
Independent
S. Bryant
Futurewei Technologies
October 27, 2019

DetNet Data Plane: MPLS over IEEE 802.1 Time Sensitive Networking (TSN)
draft-ietf-detnet-mpls-over-tsn-01

Abstract

This document specifies the Deterministic Networking MPLS data plane when operating over a TSN sub-network.

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 April 29, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Terminology	3
2.1. Terms Used in This Document	3
2.2. Abbreviations	3
2.3. Requirements Language	4
3. DetNet MPLS Data Plane Overview	4
4. DetNet MPLS Operation Over IEEE 802.1 TSN Sub-Networks	5
4.1. Functions for DetNet Flow to TSN Stream Mapping	7
4.2. TSN requirements of MPLS DetNet nodes	7
4.3. Service protection within the TSN sub-network	9
4.4. Aggregation during DetNet flow to TSN Stream mapping	9
5. Management and Control Implications	9
6. Security Considerations	11
7. IANA Considerations	11
8. Acknowledgements	11
9. References	11
9.1. Normative References	11
9.2. Informative References	11
Authors' Addresses	13

1. Introduction

Deterministic Networking (DetNet) is a service that can be offered by a network to DetNet flows. DetNet provides these flows with a low packet loss rates and assured maximum end-to-end delivery latency. General background and concepts of DetNet can be found in [I-D.ietf-detnet-architecture].

The DetNet Architecture decomposes the DetNet related data plane functions into two sub-layers: a service sub-layer and a forwarding sub-layer. The service sub-layer is used to provide DetNet service protection and reordering. The forwarding sub-layer is used to provides congestion protection (low loss, assured latency, and limited reordering) leveraging MPLS Traffic Engineering mechanisms.

[I-D.ietf-detnet-mpls] specifies the DetNet data plane operation for MPLS-based Packet Switched Network (PSN). MPLS encapsulated DetNet flows can be carried over network technologies that can provide the DetNet required level of service. This document focuses on the scenario where MPLS (DetNet) nodes are interconnected by a IEEE 802.1 TSN sub-network.

2. Terminology

[Editor's note: Needs clean up.].

2.1. Terms Used in This Document

This document uses the terminology established in the DetNet architecture [I-D.ietf-detnet-architecture] and [I-D.ietf-detnet-mpls], and the reader is assumed to be familiar with that document and its terminology.

2.2. Abbreviations

The following abbreviations are used in this document:

CW	Control Word.
DetNet	Deterministic Networking.
DF	DetNet Flow.
FRER	Frame Replication and Elimination for Redundancy (TSN function).
L2	Layer 2.
L3	Layer 3.
LSR	Label Switching Router.
MPLS	Multiprotocol Label Switching.
PE	Provider Edge.
PREOF	Packet Replication, Elimination and Ordering Functions.
PSN	Packet Switched Network.
PW	PseudoWire.
S-PE	Switching Provider Edge.
T-PE	Terminating Provider Edge.
TSN	Time-Sensitive Network.

2.3. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. DetNet MPLS Data Plane Overview

The basic approach defined in [I-D.ietf-detnet-mpls] supports the DetNet service sub-layer based on existing pseudowire (PW) encapsulations and mechanisms, and supports the DetNet forwarding sub-layer based on existing MPLS Traffic Engineering encapsulations and mechanisms.

A node operating on a DetNet flow in the Detnet service sub-layer, i.e. a node processing a DetNet packet which has the S-Label as top of stack uses the local context associated with that S-Label, for example a received F-Label, to determine what local DetNet operation(s) are applied to that packet. An S-Label may be unique when taken from the platform label space [RFC3031], which would enable correct DetNet flow identification regardless of which input interface or LSP the packet arrives on. The service sub-layer functions (i.e., PREOF) use a DetNet control word (d-CW).

The DetNet MPLS data plane builds on MPLS Traffic Engineering encapsulations and mechanisms to provide a forwarding sub-layer that is responsible for providing resource allocation and explicit routes. The forwarding sub-layer is supported by one or more forwarding labels (F-Labels).

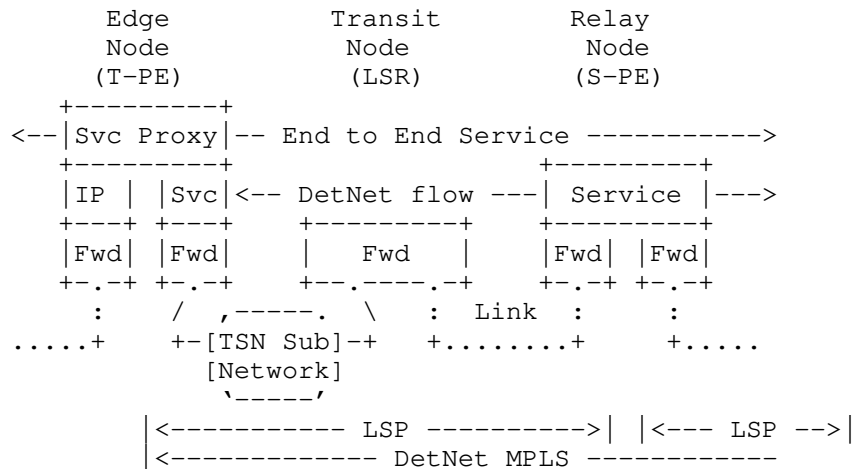


Figure 1: Part of a Simple DetNet MPLS Network using a TSN sub-net

Figure 1 illustrates an extract of a DetNet enabled MPLS network. Edge/relay nodes sit at MPLS LSP boundaries and transit nodes are LSRs. In this figure, two MPLS nodes (the edge node and the transit node) are interconnected by a TSN sub-network.

DetNet edge/relay nodes are DetNet service sub-layer aware, understand the particular needs of DetNet flows and provide both DetNet service and forwarding sub-layer functions. They add, remove and process d-CWs, S-Labels and F-labels as needed. MPLS enabled DetNet nodes can enhance the reliability of delivery by enabling the replication of packets where multiple copies, possibly over multiple paths, are forwarded through the DetNet domain. They can also eliminate surplus previously replicated copies of DetNet packets. MPLS (DetNet) nodes also include DetNet forwarding sub-layer functions, support for notably explicit routes, and resources allocation to eliminate (or reduce) congestion loss and jitter.

DetNet transit nodes reside wholly within a DetNet domain, and also provide DetNet forwarding sub-layer functions in accordance with the performance required by a DetNet flow carried over an LSP. Unlike other DetNet node types, transit nodes provide no service sub-layer processing.

4. DetNet MPLS Operation Over IEEE 802.1 TSN Sub-Networks

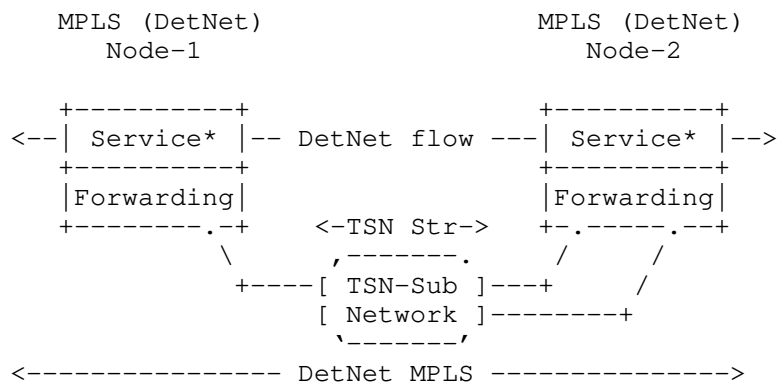
The DetNet WG collaborates with IEEE 802.1 TSN in order to define a common architecture for both Layer 2 and Layer 3, what maintains consistency across diverse networks. Both DetNet MPLS and TSN use the same techniques to provide their deterministic service:

- o Service protection.
- o Resource allocation.
- o Explicit routes.

As described in the DetNet architecture [I-D.ietf-detnet-architecture] and also illustrated here in Figure 1 a sub-network provides from MPLS perspective a single hop connection between MPLS (DetNet) nodes. Functions used for resource allocation and explicit routes are treated as domain internal functions and does not require function interworking across the DetNet MPLS network and the TSN sub-network.

In case of the service protection function due to the similarities of the DetNet PREOF and TSN FRER functions some level of interworking is possible. However, such interworking is out-of-scope in this document and left for further study.

Figure 2 illustrates a scenario, where two MPLS (DetNet) nodes are interconnected by a TSN sub-network. Node-1 is single homed and Node-2 is dual-homed to the TSN sub-network.



Note: * no service sub-layer required for transit nodes

Figure 2: DetNet Enabled MPLS Network Over a TSN Sub-Network

The Time-Sensitive Networking (TSN) Task Group of the IEEE 802.1 Working Group have defined (and are defining) a number of amendments to IEEE 802.1Q [IEEE8021Q] that provide zero congestion loss and bounded latency in bridged networks. Furthermore IEEE 802.1CB [IEEE8021CB] defines frame replication and elimination functions for reliability that should prove both compatible with and useful to,

DetNet networks. All these functions have to identify flows those require TSN treatment.

TSN capabilities of the TSN sub-network are made available for MPLS (DetNet) flows via the protocol interworking function defined in IEEE 802.1CB [IEEE8021CB]. For example, applied on the TSN edge port it can convert an ingress unicast MPLS (DetNet) flow to use a specific Layer-2 multicast destination MAC address and a VLAN, in order to direct the packet through a specific path inside the bridged network. A similar interworking function pair at the other end of the TSN sub-network would restore the packet to its original Layer-2 destination MAC address and VLAN.

Placement of TSN functions depends on the TSN capabilities of nodes. MPLS (DetNet) Nodes may or may not support TSN functions. For a given TSN Stream (i.e., DetNet flow) an MPLS (DetNet) node is treated as a Talker or a Listener inside the TSN sub-network.

4.1. Functions for DetNet Flow to TSN Stream Mapping

Mapping of a DetNet MPLS flow to a TSN Stream is provided via the combination of a passive and an active stream identification function that operate at the frame level. The passive stream identification function is used to catch the MPLS label(s) of a DetNet MPLS flow and the active stream identification function is used to modify the Ethernet header according to the ID of the mapped TSN Stream.

IEEE P802.1CBdb [IEEEP8021CBdb] defines a Mask-and-Match Stream identification function that can be used as a passive function for MPLS DetNet flows.

IEEE 802.1CB [IEEE8021CB] defines an Active Destination MAC and VLAN Stream identification function, what can replace some Ethernet header fields namely (1) the destination MAC-address, (2) the VLAN-ID and (3) priority parameters with alternate values. Replacement is provided for the frame passed down the stack from the upper layers or up the stack from the lower layers.

Active Destination MAC and VLAN Stream identification can be used within a Talker to set flow identity or a Listener to recover the original addressing information. It can be used also in a TSN bridge that is providing translation as a proxy service for an End System.

4.2. TSN requirements of MPLS DetNet nodes

This section covers required behavior of a TSN-aware MPLS (DetNet) node using a TSN sub-network.

From the TSN sub-network perspective MPLS (DetNet) nodes are treated as Talker or Listener, that may be (1) TSN-unaware or (2) TSN-aware.

In cases of TSN-unaware MPLS DetNet nodes the TSN relay nodes within the TSN sub-network must modify the Ethernet encapsulation of the DetNet MPLS flow (e.g., MAC translation, VLAN-ID setting, Sequence number addition, etc.) to allow proper TSN specific handling inside the sub-network. There are no requirements defined for TSN-unaware MPLS DetNet nodes in this document.

MPLS (DetNet) nodes being TSN-aware can be treated as a combination of a TSN-unaware Talker/Listener and a TSN-Relay, as shown in Figure 3. In such cases the MPLS (DetNet) node must provide the TSN sub-network specific Ethernet encapsulation over the link(s) towards the sub-network.

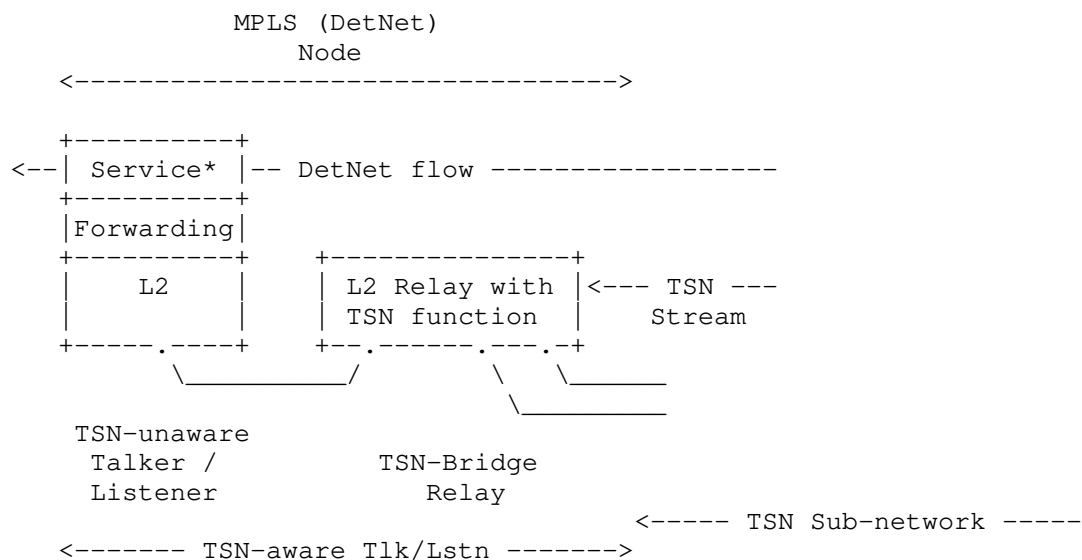


Figure 3: MPLS (DetNet) Node with TSN Functions

A TSN-aware MPLS (DetNet) node implementations MUST support the Stream Identification TSN component for recognizing flows.

A Stream identification component MUST be able to instantiate the following functions (1) Active Destination MAC and VLAN Stream identification function, (2) Mask-and-Match Stream identification function and (3) the related managed objects in Clause 9 of IEEE 802.1CB [IEEE8021CB] and IEEE P802.1CBdb [IEEEP8021CBdb].

A TSN-aware MPLS (DetNet) node implementations MUST support the Sequencing function and the Sequence encode/decode function as defined in IEEE 802.1CB [IEEE8021CB] if FRER is used inside the TSN sub-network.

The Sequence encode/decode function MUST support the Redundancy tag (R-TAG) format as per Clause 7.8 of IEEE 802.1CB [IEEE8021CB].

A TSN-aware MPLS (DetNet) node implementations MUST support the Stream splitting function and the Individual recovery function as defined in IEEE 802.1CB [IEEE8021CB] when the node is a replication or elimination point for FRER.

4.3. Service protection within the TSN sub-network

TSN Streams supporting DetNet flows may use Frame Replication and Elimination for Redundancy (FRER) as defined in IEEE 802.1CB [IEEE8021CB] based on the loss service requirements of the TSN Stream, which is derived from the DetNet service requirements of the DetNet mapped flow. The specific operation of FRER is not modified by the use of DetNet and follows IEEE 802.1CB [IEEE8021CB].

FRER function and the provided service recovery is available only within the TSN sub-network as the TSN Stream-ID and the TSN sequence number are not valid outside the sub-network. An MPLS (DetNet) node represents a L3 border and as such it terminates all related information elements encoded in the L2 frames.

As the Stream-ID and the TSN sequence number are paired with the similar MPLS flow parameters, FRER can be combined with PREOF functions. Such service protection interworking scenarios may require to move sequence number fields among TSN (L2) and PW (MPLS) encapsulations and they are left for further study.

4.4. Aggregation during DetNet flow to TSN Stream mapping

Implementations of this document SHALL use management and control information to map a DetNet flow to a TSN Stream. N:1 mapping (aggregating DetNet flows in a single TSN Stream) SHALL be supported. The management or control function that provisions flow mapping SHALL ensure that adequate resources are allocated and configured to provide proper service requirements of the mapped flows.

5. Management and Control Implications

[Editor's note: This section covers management/control plane related implications of creation, mapping, removal of TSN Stream IDs, their related parameters and, when needed, the configuration of FRER.]

DetNet flow and TSN Stream mapping related information are required only for TSN-aware MPLS (DetNet) nodes. From the Data Plane perspective there is no practical difference based on the origin of flow mapping related information (management plane or control plane).

TSN-aware MPLS DetNet nodes are member of both the DetNet domain and the TSN sub-network. Within the TSN sub-network the TSN-aware MPLS (DetNet) node has a TSN-aware Talker/Listener role, so TSN specific management and control plane functionalities must be implemented. There are many similarities in the management plane techniques used in DetNet and TSN, but that is not the case for the control plane protocols. For example, RSVP-TE and MSRP behaves differently. Therefore management and control plane design is an important aspect of scenarios, where mapping between DetNet and TSN is required.

In order to use a TSN sub-network between DetNet nodes, DetNet specific information must be converted to TSN sub-network specific ones. DetNet flow ID and flow related parameters/requirements must be converted to a TSN Stream ID and stream related parameters/requirements. Note that, as the TSN sub-network is just a portion of the end2end DetNet path (i.e., single hop from MPLS perspective), some parameters (e.g., delay) may differ significantly. Other parameters (like bandwidth) also may have to be tuned due to the L2 encapsulation used within the TSN sub-network.

In some case it may be challenging to determine some TSN Stream related information. For example, on a TSN-aware MPLS (DetNet) node that acts as a Talker, it is quite obvious which DetNet node is the Listener of the mapped TSN stream (i.e., the MPLS Next-Hop). However it may be not trivial to locate the point/interface where that Listener is connected to the TSN sub-network. Such attributes may require interaction between control and management plane functions and between DetNet and TSN domains.

Mapping between DetNet flow identifiers and TSN Stream identifiers, if not provided explicitly, can be done by a TSN-aware MPLS (DetNet) node locally based on information provided for configuration of the TSN Stream identification functions (Mask-and-match Stream identification and active Stream identification function).

Triggering the setup/modification of a TSN Stream in the TSN sub-network is an example where management and/or control plane interactions are required between the DetNet and TSN sub-network. TSN-unaware MPLS (DetNet) nodes make such a triggering even more complicated as they are fully unaware of the sub-network and run independently.

Configuration of TSN specific functions (e.g., FRER) inside the TSN sub-network is a TSN domain specific decision and may not be visible in the DetNet domain. Service protection interworking scenarios are left for further study.

6. Security Considerations

The security considerations of DetNet in general are discussed in [I-D.ietf-detnet-architecture] and [I-D.ietf-detnet-security]. DetNet IP data plane specific considerations are summarized in [I-D.ietf-detnet-ip]. Encryption may be provided by an underlying sub-net using MACSec [IEEE802.1AE-2018] for DetNet IP over TSN flows.

7. IANA Considerations

This document makes no IANA requests.

8. Acknowledgements

The authors wish to thank Norman Finn, Lou Berger, Craig Gunther, Christophe Mangin and Jouni Korhonen for their various contributions to this work.

9. References

9.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC3031] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, DOI 10.17487/RFC3031, January 2001, <<https://www.rfc-editor.org/info/rfc3031>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

9.2. Informative References

- [G.8275.1]
International Telecommunication Union, "Precision time protocol telecom profile for phase/time synchronization with full timing support from the network", ITU-T G.8275.1/Y.1369.1 G.8275.1, June 2016, <<https://www.itu.int/rec/T-REC-G.8275.1/en>>.
- [G.8275.2]
International Telecommunication Union, "Precision time protocol telecom profile for phase/time synchronization with partial timing support from the network", ITU-T G.8275.2/Y.1369.2 G.8275.2, June 2016, <<https://www.itu.int/rec/T-REC-G.8275.2/en>>.
- [I-D.ietf-detnet-architecture]
Finn, N., Thubert, P., Varga, B., and J. Farkas, "Deterministic Networking Architecture", draft-ietf-detnet-architecture-13 (work in progress), May 2019.
- [I-D.ietf-detnet-ip]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: IP", draft-ietf-detnet-ip-01 (work in progress), July 2019.
- [I-D.ietf-detnet-mpls]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: MPLS", draft-ietf-detnet-mpls-01 (work in progress), July 2019.
- [I-D.ietf-detnet-security]
Mizrahi, T., Grossman, E., Hacker, A., Das, S., Dowdell, J., Austad, H., Stanton, K., and N. Finn, "Deterministic Networking (DetNet) Security Considerations", draft-ietf-detnet-security-05 (work in progress), August 2019.
- [IEEE1588]
IEEE, "IEEE 1588 Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems Version 2", 2008.
- [IEEE802.1AE-2018]
IEEE Standards Association, "IEEE Std 802.1AE-2018 MAC Security (MACsec)", 2018, <<https://ieeexplore.ieee.org/document/8585421>>.

[IEEE8021CB]

Finn, N., "Draft Standard for Local and metropolitan area networks - Seamless Redundancy", IEEE P802.1CB /D2.1 P802.1CB, December 2015, <<http://www.ieee802.org/1/files/private/cb-drafts/d2/802-1CB-d2-1.pdf>>.

[IEEE8021Q]

IEEE 802.1, "Standard for Local and metropolitan area networks--Bridges and Bridged Networks (IEEE Std 802.1Q-2014)", 2014, <<http://standards.ieee.org/about/get/>>.

[IEEEP8021CBdb]

Mangin, C., "Extended Stream identification functions", IEEE P802.1CBdb /D0.2 P802.1CBdb, August 2019, <<http://www.ieee802.org/1/files/private/cb-drafts/d2/802-1CB-d2-1.pdf>>.

Authors' Addresses

Balazs Varga (editor)
Ericsson
Magyar Tudosok krt. 11.
Budapest 1117
Hungary

Email: balazs.a.varga@ericsson.com

Janos Farkas
Ericsson
Magyar Tudosok krt. 11.
Budapest 1117
Hungary

Email: janos.farkas@ericsson.com

Andrew G. Malis
Independent

Email: agmalis@gmail.com

Stewart Bryant
Futurewei Technologies

Email: stewart.bryant@gmail.com

DetNet
Internet-Draft
Intended status: Standards Track
Expires: April 29, 2020

B. Varga, Ed.
J. Farkas
Ericsson
L. Berger
LabN Consulting, L.L.C.
A. Malis
Independent
S. Bryant
Futurewei Technologies
J. Korhonen
October 27, 2019

DetNet Data Plane: MPLS over UDP/IP
draft-ietf-detnet-mpls-over-udp-ip-03

Abstract

This document specifies the MPLS Deterministic Networking data plane operation and encapsulation over an IP network. The approach is modeled on the operation of MPLS and over UDP/IP packet switched networks.

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 April 29, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Terminology	3
2.1. Terms Used in This Document	3
2.2. Abbreviations	3
2.3. Requirements Language	4
3. DetNet MPLS Operation over DetNet	
IP PSNs	4
4. DetNet Data Plane Procedures	5
5. Management and Control Information Summary	6
6. Security Considerations	6
7. IANA Considerations	6
8. Acknowledgements	6
9. References	7
9.1. Normative References	7
9.2. Informative References	7
Authors' Addresses	8

1. Introduction

Deterministic Networking (DetNet) is a service that can be offered by a network to DetNet flows. DetNet provides these flows extremely low packet loss rates and assured maximum end-to-end delivery latency. General background and concepts of DetNet can be found in [RFC8655].

This document specifies use of the MPLS DetNet encapsulation over an IP network. The approach is modeled on the operation of MPLS over an IP Packet Switched Network (PSN) [RFC7510]. It maps the MPLS data plane encapsulation described in [I-D.ietf-detnet-mpls] to the DetNet IP data plane defined in [I-D.ietf-detnet-ip].

To carry DetNet MPLS flows with full functionality at the DetNet layer over an IP network, the following components are required (these are a subset of the requirements for MPLS encapsulation listed in [I-D.ietf-detnet-mpls]):

1. A method for identifying DetNet flows to the processing element.
2. A method for carrying the DetNet sequence number.

3. A method for distinguishing DetNet OAM packets from DetNet data packets.
4. A method for carrying queuing and forwarding indication.

These requirements are satisfied by the DetNet over MPLS Encapsulation described in [I-D.ietf-detnet-mpls] and they are partly satisfied by the DetNet IP data plane defined in [I-D.ietf-detnet-ip]

2. Terminology

2.1. Terms Used in This Document

This document uses the terminology established in the DetNet architecture [RFC8655], and the reader is assumed to be familiar with that document and its terminology.

2.2. Abbreviations

The following abbreviations are used in this document:

d-CW	A DetNet Control Word (d-CW) is used for sequencing and identifying duplicate packets of a DetNet flow at the DetNet service sub-layer.
DetNet	Deterministic Networking.
A-Label	A special case of an S-Label, whose properties are known only at the aggregation and deaggregation end-points.
F-Label	A Detnet "forwarding" label that identifies the LSP used to forward a DetNet flow across an MPLS PSN, e.g., a hop-by-hop label used between label switching routers.
MPLS	Multiprotocol Label Switching.
OAM	Operations, Administration, and Maintenance.
PEF	Packet Elimination Function.
POF	Packet Ordering Function.
PREOF	Packet Replication, Elimination and Ordering Functions.
PRF	Packet Replication Function.

PSN	Packet Switched Network.
S-Label	A DetNet "service" label that is used between DetNet nodes that also implement the DetNet service sub-layer functions. An S-Label is also used to identify a DetNet flow at DetNet service sub-layer.

2.3. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. DetNet MPLS Operation over DetNet IP PSNs

This document builds on the specification of MPLS over UDP defined in [RFC7510]. It may replace partly or entirely the F-Label(s) used in [I-D.ietf-detnet-mpls] with UDP and IP headers. The UDP and IP header information is used to identify DetNet flows, including member flows, per [I-D.ietf-detnet-ip]. The resulting encapsulation is shown in Figure 1. There may be zero or more F-label(s) between the S-label and the UDP header.

Note that this encapsulation works equally well with IPv4, IPv6, and IPv6-based Segment Routing [I-D.ietf-6man-segment-routing-header].

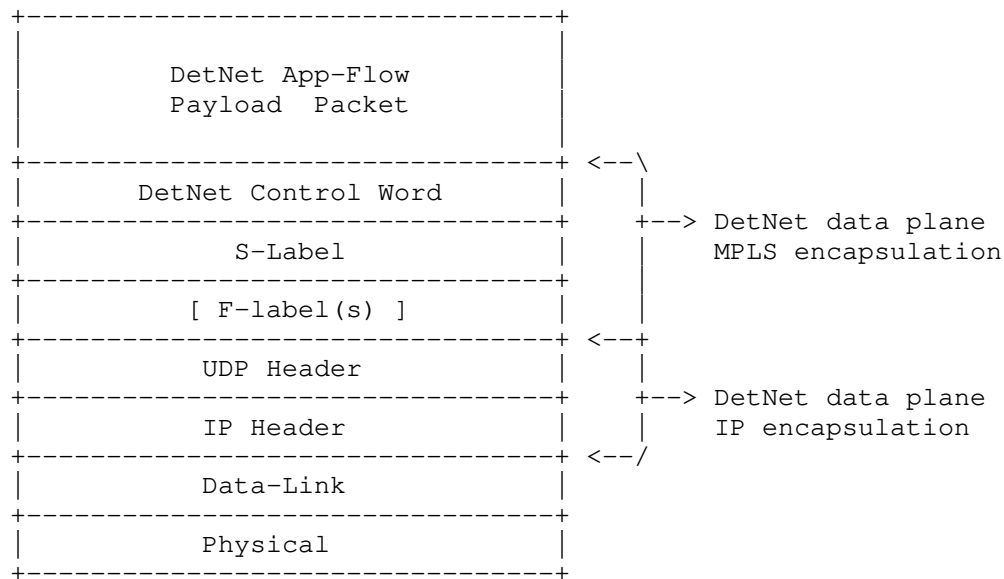


Figure 1: UDP/IP Encapsulation of DetNet MPLS

S-Labels, d-CW and zero or more F-Labels are used as defined in [I-D.ietf-detnet-mpls] and are not modified by this document. In case of aggregates the A-Label is treated as an S-Label and it too is not modified.

4. DetNet Data Plane Procedures

To support outgoing DetNet MPLS over UDP/IP encapsulation, an implementation MUST support the provisioning of UDP and IP header information in addition or in place of F-Label(s). Note, when PRF is performed at the MPLS service sub-layer, there will be multiple member flows, and each member flow will require the provisioning of their own UDP and IP header information. The headers for each outgoing packet MUST be formatted according to the configuration information and as defined in [RFC7510], with one exception. Note that the UDP Source Port value MUST be set to uniquely identify the DetNet flow. The packet MUST then be handed as a DetNet IP packet, per [I-D.ietf-detnet-ip]. This includes QoS related traffic treatment.

To support receive processing an implementation MUST also support the provisioning of received UDP and IP header information. The provisioned information MUST be used to identify incoming app-flows based on the combination of S-Label and incoming encapsulation header

information. Normal receive processing as defined in [I-D.ietf-detnet-mpls], including PEF and POF, can then take place.

5. Management and Control Information Summary

The following summarizes the set of information that is needed to configure DetNet MPLS over UDP/IP:

- o Label information (S-label or F-label) to be mapped to UDP/IP flow. Note that a single S-Label can map to multiple sets of UDP/IP information when PREOF is used.
- o IPv4 or IPv6 source address field.
- o IPv4 or IPv6 destination address field.
- o IPv4 Type of Service or IPv6 Traffic Class Fields.
- o UDP Source Port.
- o UDP Destination Port.

This information MUST be provisioned per DetNet flow via configuration, e.g., via the controller or management plane.

It is the responsibility of the DetNet controller plane to properly provision both flow identification information and the flow specific resources needed to provide the traffic treatment needed to meet each flow's service requirements. This applies for aggregated and individual flows.

6. Security Considerations

The security considerations of DetNet in general are discussed in [RFC8655] and [I-D.ietf-detnet-security]. MPLS and IP specific security considerations are described in [I-D.ietf-detnet-mpls] and [I-D.ietf-detnet-ip]. This draft does not have additional security considerations.

7. IANA Considerations

This document makes no IANA requests.

8. Acknowledgements

The authors wish to thank Pat Thaler, Norman Finn, Loa Anderson, David Black, Rodney Cummings, Ethan Grossman, Tal Mizrahi, David

Mozes, Craig Gunther, George Swallow, Yuanlong Jiang and Carlos J. Bernardos for their various contributions to this work.

9. References

9.1. Normative References

- [I-D.ietf-detnet-ip]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: IP", draft-ietf-detnet-ip-01 (work in progress), July 2019.
- [I-D.ietf-detnet-mpls]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: MPLS", draft-ietf-detnet-mpls-01 (work in progress), July 2019.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC7510] Xu, X., Sheth, N., Yong, L., Callon, R., and D. Black, "Encapsulating MPLS in UDP", RFC 7510, DOI 10.17487/RFC7510, April 2015, <<https://www.rfc-editor.org/info/rfc7510>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

9.2. Informative References

- [I-D.ietf-6man-segment-routing-header]
Filsfils, C., Dukes, D., Previdi, S., Leddy, J., Matsushima, S., and d. daniel.voyer@bell.ca, "IPv6 Segment Routing Header (SRH)", draft-ietf-6man-segment-routing-header-26 (work in progress), October 2019.
- [I-D.ietf-detnet-security]
Mizrahi, T., Grossman, E., Hacker, A., Das, S., Dowdell, J., Austad, H., Stanton, K., and N. Finn, "Deterministic Networking (DetNet) Security Considerations", draft-ietf-detnet-security-05 (work in progress), August 2019.

[RFC8655] Finn, N., Thubert, P., Varga, B., and J. Farkas,
"Deterministic Networking Architecture", RFC 8655,
DOI 10.17487/RFC8655, October 2019,
<<https://www.rfc-editor.org/info/rfc8655>>.

Authors' Addresses

Balazs Varga (editor)
Ericsson
Magyar Tudosok krt. 11.
Budapest 1117
Hungary

Email: balazs.a.varga@ericsson.com

Janos Farkas
Ericsson
Magyar Tudosok krt. 11.
Budapest 1117
Hungary

Email: janos.farkas@ericsson.com

Lou Berger
LabN Consulting, L.L.C.

Email: lberger@labn.net

Andrew G. Malis
Independent

Email: agmalis@gmail.com

Stewart Bryant
Futurewei Technologies

Email: stewart.bryant@gmail.com

Jouni Korhonen

Email: jouni.nospam@gmail.com

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: May 4, 2020

T. Mizrahi
HUAWEI
E. Grossman, Ed.
DOLBY
A. Hacker
MISTIQ
S. Das
Applied Communication Sciences
J. Dowdell
Airbus Defence and Space
H. Austad
SINTEF Digital
N. Finn
HUAWEI
November 1, 2019

Deterministic Networking (DetNet) Security Considerations
draft-ietf-detnet-security-06

Abstract

A deterministic network is one that can carry data flows for real-time applications with extremely low data loss rates and bounded latency. Deterministic networks have been successfully deployed in real-time operational technology (OT) applications for some years. However, such networks are typically isolated from external access, and thus the security threat from external attackers is low. IETF Deterministic Networking (DetNet) specifies a set of technologies that enable creation of deterministic networks on IP-based networks of potentially wide area (on the scale of a corporate network) potentially bringing the OT network into contact with Information Technology (IT) traffic and security threats that lie outside of a tightly controlled and bounded area (such as the internals of an aircraft). These DetNet technologies have not previously been deployed together on a wide area IP-based network, and thus can present security considerations that may be new to IP-based wide area network designers. This draft, intended for use by DetNet network designers, provides insight into these security considerations. In addition, this draft collects all security-related statements from the various DetNet drafts (Architecture, Use Cases, etc) into a single location Section 8.

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 May 4, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	4
2. Abbreviations	5
3. Security Threats	6
3.1. Threat Model	6
3.2. Threat Analysis	7
3.2.1. Delay	7
3.2.1.1. Delay Attack	7
3.2.2. DetNet Flow Modification or Spoofing	7
3.2.3. Resource Segmentation or Slicing	7
3.2.3.1. Inter-segment Attack	8
3.2.4. Packet Replication and Elimination	8
3.2.4.1. Replication: Increased Attack Surface	8
3.2.4.2. Replication-related Header Manipulation	8
3.2.5. Path Choice	9
3.2.5.1. Path Manipulation	9
3.2.5.2. Path Choice: Increased Attack Surface	9
3.2.6. Control Plane	9
3.2.6.1. Control or Signaling Packet Modification	9
3.2.6.2. Control or Signaling Packet Injection	9

3.2.7.	Scheduling or Shaping	9
3.2.7.1.	Reconnaissance	9
3.2.8.	Time Synchronization Mechanisms	9
3.3.	Threat Summary	10
4.	Security Threat Impacts	10
4.1.	Delay-Attacks	13
4.1.1.	Data Plane Delay Attacks	13
4.1.2.	Control Plane Delay Attacks	14
4.2.	Flow Modification and Spoofing	14
4.2.1.	Flow Modification	14
4.2.2.	Spoofing	14
4.2.2.1.	Dataplane Spoofing	14
4.2.2.2.	Control Plane Spoofing	14
4.3.	Segmentation attacks (injection)	15
4.3.1.	Data Plane Segmentation	15
4.3.2.	Control Plane segmentation	15
4.4.	Replication and Elimination	15
4.4.1.	Increased Attack Surface	16
4.4.2.	Header Manipulation at Elimination Bridges	16
4.5.	Control or Signaling Packet Modification	16
4.6.	Control or Signaling Packet Injection	16
4.7.	Reconnaissance	16
4.8.	Attacks on Time Sync Mechanisms	16
4.9.	Attacks on Path Choice	16
5.	Security Threat Mitigation	16
5.1.	Path Redundancy	17
5.2.	Integrity Protection	17
5.3.	DetNet Node Authentication	18
5.4.	Dummy Traffic Insertion	18
5.5.	Encryption	18
5.5.1.	Encryption Considerations for DetNet	19
5.6.	Control and Signaling Message Protection	20
5.7.	Dynamic Performance Analytics	20
5.8.	Mitigation Summary	21
6.	Association of Attacks to Use Cases	22
6.1.	Use Cases by Common Themes	22
6.1.1.	Network Layer - AVB/TSN Ethernet	22
6.1.2.	Central Administration	23
6.1.3.	Hot Swap	23
6.1.4.	Data Flow Information Models	24
6.1.5.	L2 and L3 Integration	24
6.1.6.	End-to-End Delivery	24
6.1.7.	Proprietary Deterministic Ethernet Networks	25
6.1.8.	Replacement for Proprietary Fieldbuses	25
6.1.9.	Deterministic vs Best-Effort Traffic	25
6.1.10.	Deterministic Flows	26
6.1.11.	Unused Reserved Bandwidth	26
6.1.12.	Interoperability	27

6.1.13. Cost Reductions	27
6.1.14. Insufficiently Secure Devices	27
6.1.15. DetNet Network Size	27
6.1.16. Multiple Hops	28
6.1.17. Level of Service	28
6.1.18. Bounded Latency	29
6.1.19. Low Latency	29
6.1.20. Bounded Jitter (Latency Variation)	29
6.1.21. Symmetrical Path Delays	29
6.1.22. Reliability and Availability	30
6.1.23. Redundant Paths	30
6.1.24. Security Measures	30
6.2. Attack Types by Use Case Common Theme	31
6.3. Security Considerations for OAM Traffic	33
7. DetNet Technology-Specific Threats	33
7.1. IP	34
7.2. MPLS	34
7.3. TSN	35
8. Appendix A: DetNet Draft Security-Related Statements	35
8.1. Architecture (draft 8)	35
8.1.1. Fault Mitigation (sec 4.5)	35
8.1.2. Security Considerations (sec 7)	36
8.2. Data Plane Alternatives (draft 4)	36
8.2.1. Security Considerations (sec 7)	36
8.3. Problem Statement (draft 5)	37
8.3.1. Security Considerations (sec 5)	37
8.4. Use Cases (draft 11)	37
8.4.1. (Utility Networks) Security Current Practices and Limitations (sec 3.2.1)	37
8.4.2. (Utility Networks) Security Trends in Utility Networks (sec 3.3.3)	39
8.4.3. (BAS) Security Considerations (sec 4.2.4)	41
8.4.4. (6TiSCH) Security Considerations (sec 5.3.3)	41
8.4.5. (Cellular radio) Security Considerations (sec 6.1.5)	41
8.4.6. (Industrial M2M) Communication Today (sec 7.2)	42
9. IANA Considerations	42
10. Security Considerations	42
11. Informative References	42
Authors' Addresses	45

1. Introduction

Security is of particularly high importance in DetNet networks because many of the use cases which are enabled by DetNet [RFC8578] include control of physical devices (power grid components, industrial controls, building controls) which can have high operational costs for failure, and present potentially attractive targets for cyber-attackers.

This situation is even more acute given that one of the goals of DetNet is to provide a "converged network", i.e. one that includes both IT traffic and OT traffic, thus exposing potentially sensitive OT devices to attack in ways that were not previously common (usually because they were under a separate control system or otherwise isolated from the IT network, for example [ARINC664P7]). Security considerations for OT networks is not a new area, and there are many OT networks today that are connected to wide area networks or the Internet; this draft focuses on the issues that are specific to the DetNet technologies and use cases.

The DetNet technologies include ways to:

- o Reserve data plane resources for DetNet flows in some or all of the intermediate nodes (e.g. bridges or routers) along the path of the flow
- o Provide explicit routes for DetNet flows that do not rapidly change with the network topology
- o Distribute data from DetNet flow packets over time and/or space to ensure delivery of each packet's data' in spite of the loss of a path

This draft includes sections on threat modeling and analysis, threat impact and mitigation, and the association of attacks with use cases based on the Use Case Common Themes section of the DetNet Use Cases draft [RFC8578].

This draft also provides context for the DetNet security considerations by collecting into one place Section 8 the various remarks about security from the various DetNet drafts (Use Cases, Architecture, etc). This text is duplicated here primarily because the DetNet working group has elected not to produce a Requirements draft and thus collectively these statements are as close as we have to "DetNet Security Requirements".

2. Abbreviations

IT Information technology (the application of computers to store, study, retrieve, transmit, and manipulate data or information, often in the context of a business or other enterprise - Wikipedia).

OT Operational Technology (the hardware and software dedicated to detecting or causing changes in physical processes through direct monitoring and/or control of physical devices such as valves, pumps, etc. - Wikipedia)

MITM Man in the Middle

SN Sequence Number

STRIDE Addresses risk and severity associated with threat categories: Spoofing identity, Tampering with data, Repudiation, Information disclosure, Denial of service, Elevation of privilege.

DREAD Compares and prioritizes risk represented by these threat categories: Damage potential, Reproducibility, Exploitability, how many Affected users, Discoverability.

PTP Precision Time Protocol [IEEE1588]

3. Security Threats

This section presents a threat model, and analyzes the possible threats in a DetNet-enabled network. The threats considered in this section are independent of any specific technologies used to implement the DetNet; Section 7) considers attacks that are associated with the DetNet technologies encompassed by [I-D.ietf-detnet-data-plane-framework].

We distinguish control plane threats from data plane threats. The attack surface may be the same, but the types of attacks as well as the motivation behind them, are different. For example, a delay attack is more relevant to data plane than to control plane. There is also a difference in terms of security solutions: the way you secure the data plane is often different than the way you secure the control plane.

3.1. Threat Model

The threat model used in this memo is based on the threat model of Section 3.1 of [RFC7384]. This model classifies attackers based on two criteria:

- o Internal vs. external: internal attackers either have access to a trusted segment of the network or possess the encryption or authentication keys. External attackers, on the other hand, do not have the keys and have access only to the encrypted or authenticated traffic.
- o Man in the Middle (MITM) vs. packet injector: MITM attackers are located in a position that allows interception and modification of in-flight protocol packets, whereas a traffic injector can only attack by generating protocol packets.

Care has also been taken to adhere to Section 5 of [RFC3552], both with respect to which attacks are considered out-of-scope for this document, but also which are considered to be the most common threats (explored further in Section 3.2. Most of the direct threats to DetNet are Active attacks, but it is highly suggested that DetNet application developers take appropriate measures to protect the content of the streams from passive attacks.

DetNet-Service, one of the service scenarios described in [I-D.varga-detnet-service-model], is the case where a service connects DetNet networking islands, i.e. two or more otherwise independent DetNet network domains are connected via a link that is not intrinsically part of either network. This implies that there could be DetNet traffic flowing over a non-DetNet link, which may provide an attacker with an advantageous opportunity to tamper with DetNet traffic. The security properties of non-DetNet links are outside of the scope of DetNet Security, but it should be noted that use of non-DetNet services to interconnect DetNet networks merits security analysis to ensure the integrity of the DetNet networks involved.

3.2. Threat Analysis

3.2.1. Delay

3.2.1.1. Delay Attack

An attacker can maliciously delay DetNet data flow traffic. By delaying the traffic, the attacker can compromise the service of applications that are sensitive to high delays or to high delay variation.

3.2.2. DetNet Flow Modification or Spoofing

An attacker can modify some header fields of en route packets in a way that causes the DetNet flow identification mechanisms to misclassify the flow. Alternatively, the attacker can inject traffic that is tailored to appear as if it belongs to a legitimate DetNet flow. The potential consequence is that the DetNet flow resource allocation cannot guarantee the performance that is expected when the flow identification works correctly.

3.2.3. Resource Segmentation or Slicing

3.2.3.1. Inter-segment Attack

An attacker can inject traffic, consuming network device resources, thereby affecting DetNet flows. This can be performed using non-DetNet traffic that affects DetNet traffic, or by using DetNet traffic from one DetNet flow that affects traffic from different DetNet flows.

3.2.4. Packet Replication and Elimination

3.2.4.1. Replication: Increased Attack Surface

Redundancy is intended to increase the robustness and survivability of DetNet flows, and replication over multiple paths can potentially mitigate an attack that is limited to a single path. However, the fact that packets are replicated over multiple paths increases the attack surface of the network, i.e., there are more points in the network that may be subject to attacks.

3.2.4.2. Replication-related Header Manipulation

An attacker can manipulate the replication-related header fields (R-TAG). This capability opens the door for various types of attacks. For example:

- o Forward both replicas - malicious change of a packet SN (Sequence Number) can cause both replicas of the packet to be forwarded. Note that this attack has a similar outcome to a replay attack.
- o Eliminate both replicas - SN manipulation can be used to cause both replicas to be eliminated. In this case an attacker that has access to a single path can cause packets from other paths to be dropped, thus compromising some of the advantage of path redundancy.
- o Flow hijacking - an attacker can hijack a DetNet flow with access to a single path by systematically replacing the SNs on the given path with higher SN values. For example, an attacker can replace every SN value S with a higher value $S+C$, where C is a constant integer. Thus, the attacker creates a false illusion that the attacked path has the lowest delay, causing all packets from other paths to be eliminated. Once the flow is hijacked the attacker can either replace en route packets with malicious packets, or simply injecting errors, causing the packets to be dropped at their destination.

3.2.5. Path Choice

3.2.5.1. Path Manipulation

An attacker can maliciously change, add, or remove a path, thereby affecting the corresponding DetNet flows that use the path.

3.2.5.2. Path Choice: Increased Attack Surface

One of the possible consequences of a path manipulation attack is an increased attack surface. Thus, when the attack described in the previous subsection is implemented, it may increase the potential of other attacks to be performed.

3.2.6. Control Plane

3.2.6.1. Control or Signaling Packet Modification

An attacker can maliciously modify en route control packets in order to disrupt or manipulate the DetNet path/resource allocation.

3.2.6.2. Control or Signaling Packet Injection

An attacker can maliciously inject control packets in order to disrupt or manipulate the DetNet path/resource allocation.

3.2.7. Scheduling or Shaping

3.2.7.1. Reconnaissance

A passive eavesdropper can identify DetNet flows and then gather information about en route DetNet flows, e.g., the number of DetNet flows, their bandwidths, their schedules, or other temporal properties. The gathered information can later be used to invoke other attacks on some or all of the flows.

Note that in some cases DetNet flows may be identified based on an explicit DetNet header, but in some cases the flow identification may be based on fields from the L3/L4 headers. If L3/L4 headers are involved, for purposes of this draft we assume they are encrypted and/or integrity-protected from external attackers.

3.2.8. Time Synchronization Mechanisms

An attacker can use any of the attacks described in [RFC7384] to attack the synchronization protocol, thus affecting the DetNet service.

3.3. Threat Summary

A summary of the attacks that were discussed in this section is presented in Figure 1. For each attack, the table specifies the type of attackers that may invoke the attack. In the context of this summary, the distinction between internal and external attacks is under the assumption that a corresponding security mechanism is being used, and that the corresponding network equipment takes part in this mechanism.

Attack	Attacker Type			
	Internal MITM	Inj.	External MITM	Inj.
Delay attack	+		+	
DetNet Flow Modification or Spoofing	+	+		
Inter-segment Attack	+	+		
Replication: Increased Attack Surface	+	+	+	+
Replication-related Header Manipulation	+			
Path Manipulation	+	+		
Path Choice: Increased Attack Surface	+	+	+	+
Control or Signaling Packet Modification	+			
Control or Signaling Packet Injection		+		
Reconnaissance	+		+	
Attacks on Time Sync Mechanisms	+	+	+	+

Figure 1: Threat Analysis Summary

4. Security Threat Impacts

This section describes and rates the impact of the attacks described in Section 3. In this section, the impacts as described assume that the associated mitigation is not present or has failed. Mitigations are discussed in Section 5.

In computer security, the impact (or consequence) of an incident can be measured in loss of confidentiality, integrity or availability of information.

DetNet raises these stakes significantly for OT applications, particularly those which may have been designed to run in an OT-only environment and thus may not have been designed for security in an IT environment with its associated devices, services and protocols.

The severity of various components of the impact of a successful vulnerability exploit to use cases by industry is available in more detail in [RFC8578]. Each of the use cases in the DetNet Use Cases draft is represented in the table below, including Pro Audio, Electrical Utilities, Industrial M2M (split into two areas, M2M Data Gathering and M2M Control Loop), and others.

Components of Impact (left column) include Criticality of Failure, Effects of Failure, Recovery, and DetNet Functional Dependence. Criticality of failure summarizes the seriousness of the impact. The impact of a resulting failure can affect many different metrics that vary greatly in scope and severity. In order to reduce the number of variables, only the following were included: Financial, Health and Safety, People well being (People WB), Affect on a single organization, and affect on multiple organizations. Recovery outlines how long it would take for an affected use case to get back to its pre-failure state (Recovery time objective, RTO), and how much of the original service would be lost in between the time of service failure and recovery to original state (Recovery Point Objective, RPO). DetNet dependence maps how much the following DetNet service objectives contribute to impact of failure: Time dependency, data integrity, source node integrity, availability, latency/jitter.

The scale of the Impact mappings is low, medium, and high. In some use cases there may be a multitude of specific applications in which DetNet is used. For simplicity this section attempts to average the varied impacts of different applications. This section does not address the overall risk of a certain impact which would require the likelihood of a failure happening.

In practice any such ratings will vary from case to case; the ratings shown here are given as examples.

Table, Part One (of Two)

	Pro A	Util	Bldg	Wire- less	Cell	M2M Data	M2M Ctrl

Criticality	Med	Hi	Low	Med	Med	Med	Med
Effects							
Financial	Med	Hi	Med	Med	Low	Med	Med
Health/Safety	Med	Hi	Hi	Med	Med	Med	Med
People WB	Med	Hi	Hi	Low	Hi	Low	Low
Effect 1 org	Hi	Hi	Med	Hi	Med	Med	Med
Effect >1 org	Med	Hi	Low	Med	Med	Med	Med
Recovery							
Recov Time Obj	Med	Hi	Med	Hi	Hi	Hi	Hi
Recov Point Obj	Med	Hi	Low	Med	Low	Hi	Hi
DetNet Dependence							
Time Dependency	Hi	Hi	Low	Hi	Med	Low	Hi
Latency/Jitter	Hi	Hi	Med	Med	Low	Low	Hi
Data Integrity	Hi	Hi	Med	Hi	Low	Hi	Low
Src Node Integ	Hi	Hi	Med	Hi	Med	Hi	Hi
Availability	Hi	Hi	Med	Hi	Low	Hi	Hi

Table, Part Two (of Two)

	Mining	Block Chain	Network Slicing
Criticality	Hi	Med	Hi
Effects			
Financial	Hi	Hi	Hi
Health/Safety	Hi	Low	Med
People WB	Hi	Low	Med

Effect 1 org	Hi	Hi	Hi	
+-----+	+-----+	+-----+	+-----+	+-----+
Effect >1 org	Hi	Low	Hi	
+-----+	+-----+	+-----+	+-----+	+-----+
Recovery				
+-----+	+-----+	+-----+	+-----+	+-----+
Recov Time Obj	Hi	Low	Hi	
+-----+	+-----+	+-----+	+-----+	+-----+
Recov Point Obj	Hi	Low	Hi	
+-----+	+-----+	+-----+	+-----+	+-----+
DetNet Dependence				
+-----+	+-----+	+-----+	+-----+	+-----+
Time Dependency	Hi	Low	Hi	
+-----+	+-----+	+-----+	+-----+	+-----+
Latency/Jitter	Hi	Low	Hi	
+-----+	+-----+	+-----+	+-----+	+-----+
Data Integrity	Hi	Hi	Hi	
+-----+	+-----+	+-----+	+-----+	+-----+
Src Node Integ	Hi	Hi	Hi	
+-----+	+-----+	+-----+	+-----+	+-----+
Availability	Hi	Hi	Hi	
+-----+	+-----+	+-----+	+-----+	+-----+

Figure 2: Impact of Attacks by Use Case Industry

The rest of this section will cover impact of the different groups in more detail.

4.1. Delay-Attacks

4.1.1. Data Plane Delay Attacks

Severely delayed messages in a DetNet link can result in the same behavior as dropped messages in ordinary networks as the services attached to the stream has strict deterministic requirements.

For a single path scenario, disruption is a real possibility, whereas in a multipath scenario, large delays or instabilities in one stream can lead to increased buffer and CPU resources on the elimination bridge.

A data-plane delay attack on a system controlling substantial moving devices, for example in industrial automation, can cause physical damage. For example, if the network promises a bounded latency of 2ms for a flow, yet the machine receives it with 5ms latency, the machine's control loop can become unstable.

4.1.2. Control Plane Delay Attacks

In and of itself, this is not directly a threat to the DetNet service, but the effects of delaying control messages can have quite adverse effects later.

- o Delayed tear-down can lead to resource leakage, which in turn can result in failure to allocate new streams finally giving rise to a denial of service attack.
- o Failure to deliver, or severely delaying, signalling messages adding an end-point to a multicast-group will prevent the new EP from receiving expected frames thus disrupting expected behavior.
- o Delaying messages removing an EP from a group can lead to loss of privacy as the EP will continue to receive messages even after it is supposedly removed.

4.2. Flow Modification and Spoofing

4.2.1. Flow Modification

ToDo.

4.2.2. Spoofing

4.2.2.1. Dataplane Spoofing

Spoofing dataplane messages can result in increased resource consumptions on the bridges throughout the network as it will increase buffer usage and CPU utilization. This can lead to resource exhaustion and/or increased delay.

If the attacker manages to create valid headers, the false messages can be forwarded through the network, using part of the allocated bandwidth. This in turn can cause legitimate messages to be dropped when the budget has been exhausted.

Finally, the endpoint will have to deal with invalid messages being delivered to the endpoint instead of (or in addition to) a valid message.

4.2.2.2. Control Plane Spoofing

A successful control plane spoofing-attack will potentially have adverse effects. It can do virtually anything from:

- o modifying existing streams by changing the available bandwidth

- o add or remove endpoints from a stream
- o drop streams completely
- o falsely create new streams (exhaust the systems resources, or to enable streams outside the Network engineer's control)

4.3. Segmentation attacks (injection)

4.3.1. Data Plane Segmentation

Injection of false messages in a DetNet stream could lead to exhaustion of the available bandwidth for a stream if the bridges accounts false messages to the stream's budget.

In a multipath scenario, injected messages will cause increased CPU utilization in elimination bridges. If enough paths are subject to malicious injection, the legitimate messages can be dropped. Likewise it can cause an increase in buffer usage. In total, it will consume more resources in the bridges than normal, giving rise to a resource exhaustion attack on the bridges.

If a stream is interrupted, the end application will be affected by what is now a non-deterministic stream.

4.3.2. Control Plane segmentation

A successful Control Plane segmentation attack control messages to be interpreted by nodes in the network, unbeknownst to the central controller or the network engineer. This has the potential to create

- o new streams (exhausting resources)
- o drop existing (denial of service)
- o add/remove end-stations to a multicast group (loss of privacy)
- o modify the stream attributes (affecting available bandwidth)

4.4. Replication and Elimination

The Replication and Elimination is relevant only to Data Plane messages as Signalling is not subject to multipath routing.

4.4.1. Increased Attack Surface

Covered briefly in Section 4.3

4.4.2. Header Manipulation at Elimination Bridges

Covered briefly in Section 4.3

4.5. Control or Signaling Packet Modification

ToDo.

4.6. Control or Signaling Packet Injection

ToDo.

4.7. Reconnaissance

Of all the attacks, this is one of the most difficult to detect and counter. Often, an attacker will start out by observing the traffic going through the network and use the knowledge gathered in this phase to mount future attacks.

The attacker can, at their leisure, observe over time all aspects of the messaging and signalling, learning the intent and purpose of all traffic flows. At some later date, possibly at an important time in an operational context, the attacker can launch a multi-faceted attack, possibly in conjunction with some demand for ransom.

The flow-id in the header of the data plane-messages gives an attacker a very reliable identifier for DetNet traffic, and this traffic has a high probability of going to lucrative targets.

4.8. Attacks on Time Sync Mechanisms

ToDo.

4.9. Attacks on Path Choice

This is covered in part in Section 4.3, and as with Replication and Elimination (Section 4.4, this is relevant for DataPlane messages.

5. Security Threat Mitigation

This section describes a set of measures that can be taken to mitigate the attacks described in Section 3. These mitigations should be viewed as a toolset that includes several different and

diverse tools. Each application or system will typically use a subset of these tools, based on a system-specific threat analysis.

5.1. Path Redundancy

Description

A DetNet flow that can be forwarded simultaneously over multiple paths. Path replication and elimination [RFC8655] provides resiliency to dropped or delayed packets. This redundancy improves the robustness to failures and to man-in-the-middle attacks.

Related attacks

Path redundancy can be used to mitigate various man-in-the-middle attacks, including attacks described in Section 3.2.1, Section 3.2.2, Section 3.2.3, and Section 3.2.8.

5.2. Integrity Protection

Description

An integrity protection mechanism, such as a Hash-based Message Authentication Code (HMAC) can be used to mitigate modification attacks. Integrity protection can be used on the data plane header, to prevent its modification and tampering. Integrity protection in the control plane is discussed in Section 5.6.

Packet Sequence Number Integrity Considerations

The use of PREOF in a DetNet implementation implies the use of a sequence number for each packet. There is a trust relationship between the device that adds the sequence number and the device that removes the sequence number. The sequence number may be end-to-end source to destination, or may be added/deleted by network edge devices. The adder and remover(s) have the trust relationship because they are the ones that ensure that the sequence numbers are not modifiable. Between those two points, there may or may not be replication and elimination functions. The elimination functions must be able to see the sequence numbers. Therefore any encryption that is done between adders and removers must not obscure the sequence number. If the sequence removers and the eliminators are in the same physical device, it may be possible to obscure the sequence number, however that is a layer violation, and is not recommended practice.

Related attacks

Integrity protection mitigates attacks related to modification and tampering, including the attacks described in Section 3.2.2 and Section 3.2.4.

5.3. DetNet Node Authentication

Description

Source authentication verifies the authenticity of DetNet sources, enabling mitigation of spoofing attacks. Note that while integrity protection (Section 5.2) prevents intermediate nodes from modifying information, authentication can provide traffic origin verification, i.e. to verify that each packet in a DetNet flow is from a trusted source. Authentication may be implemented as part of ingress filtering, for example.

Related attacks

DetNet node authentication is used to mitigate attacks related to spoofing, including the attacks of Section 3.2.2, and Section 3.2.4.

5.4. Dummy Traffic Insertion

Description

With some queueing methods such as [IEEE802.1Qch-2017] it is possible to introduce dummy traffic in order to regularize the timing of packet transmission.

Related attacks

Removing distinctive temporal properties of individual packets or flows can be used to mitigate against reconnaissance attacks Section 3.2.7.

5.5. Encryption

Description

DetNet flows can be forwarded in encrypted form at the DetNet layer. Alternatively, if the payload is end-to-end encrypted at the application layer, the DetNet nodes should not have any need to inspect the payload itself, and thus the DetNet implementation can be data-agnostic.

Related attacks

Encryption can be used to mitigate recon attacks (Section 3.2.7). However, for a DetNet network to give differentiated quality of service on a flow-by-flow basis, the network must be able to identify the flows individually. This implies that in a recon attack the attacker may also be able to track individual flows to learn more about the system.

5.5.1. Encryption Considerations for DetNet

Any compute time which is required for encryption and decryption processing ('crypto') must be included in the flow latency calculations. Thus, crypto algorithms used in a DetNet must have bounded worst-case execution times, and these values must be used in the latency calculations.

Some crypto algorithms are symmetric in encode/decode time (such as AES) and others are asymmetric (such as public key algorithms). There are advantages and disadvantages to the use of either type in a given DetNet context.

Asymmetrical crypto is typically not used in networks on a packet-by-packet basis due to its computational cost. For example, if only endpoint checks or checks at a small number of intermediate points are required, asymmetric crypto can be used to authenticate distribution or exchange of a secret symmetric crypto key; a successful check based on that key will provide traffic origin verification, as long as the key is kept secret by the participants. TLS and IKE (for IPsec) are examples of this for endpoint checks.

However, if secret symmetrical keys are used for this purpose the key must be given to all relays, which increases the probability of a secret key being leaked. Also, if any relay is compromised or misbehaving it may inject traffic into the flow.

Alternatively, asymmetric crypto can provide traffic origin verification at every intermediate node. For example, a DetNet flow can be associated with an (asymmetric) keypair, such that the private key is available to the source of the flow and the public key is distributed with the flow information, allowing verification at every node for every packet. However, this is more computationally expensive.

In either case, origin verification also requires replay detection as part of the security protocol to prevent an attacker from recording and resending traffic, e.g., as a denial of service attack on flow forwarding resources.

If crypto keys are to be regenerated over the duration of the flow then the time required to accomplish this must be accounted for in the latency calculations.

5.6. Control and Signaling Message Protection

Description

Control and signaling messages can be protected using authentication and integrity protection mechanisms.

Related attacks

These mechanisms can be used to mitigate various attacks on the control plane, as described in Section 3.2.6, Section 3.2.8 and Section 3.2.5.

5.7. Dynamic Performance Analytics

Description

Information about the network performance can be gathered in real-time in order to detect anomalies and unusual behavior that may be the symptom of a security attack. The gathered information can be based, for example, on per-flow counters, bandwidth measurement, and monitoring of packet arrival times. Unusual behavior or potentially malicious nodes can be reported to a management system, or can be used as a trigger for taking corrective actions. The information can be tracked by DetNet end systems and transit nodes, and exported to a management system, for example using NETCONF.

Related attacks

Performance analytics can be used to mitigate various attacks, including the ones described in Section 3.2.1 (Delay Attack), Section 3.2.3 (Resource Segmentation Attack), and Section 3.2.8 (Time Sync Attack).

For example, in the case of data plane delay attacks, one possible mitigation is to timestamp the data at the source, and timestamp it again at the destination, and if the resulting latency exceeds the promised bound, discard that data and warn the operator (and/or enter a fail-safe mode).

5.8. Mitigation Summary

The following table maps the attacks of Section 3 to the impacts of Section 4, and to the mitigations of the current section. Each row specifies an attack, the impact of this attack if it is successfully implemented, and possible mitigation methods.

Attack	Impact	Mitigations
Delay Attack	-Non-deterministic delay -Data disruption -Increased resource consumption	-Path redundancy -Performance analytics
Reconnaissance	-Enabler for other attacks	-Encryption -Dummy traffic insertion
DetNet Flow Modification or Spoofing	-Increased resource consumption -Data disruption	-Path redundancy -Integrity protection -DetNet Node authentication
Inter-Segment Attack	-Increased resource consumption -Data disruption	-Path redundancy -Performance analytics
Replication: Increased attack surface	-All impacts of other attacks	-Integrity protection -DetNet Node authentication
Replication-related Header Manipulation	-Non-deterministic delay -Data disruption	-Integrity protection -DetNet Node authentication
Path Manipulation	-Enabler for other attacks	-Control message protection
Path Choice: Increased Attack Surface	-All impacts of other attacks	-Control message protection
Control or Signaling Packet Modification	-Increased resource consumption -Non-deterministic delay	-Control message protection

	-Data disruption	
Control or Signaling Packet Injection	-Increased resource consumption -Non-deterministic delay -Data disruption	-Control message protection
Attacks on Time Sync Mechanisms	-Non-deterministic delay -Increased resource consumption -Data disruption	-Path redundancy -Control message protection -Performance analytics

Figure 3: Mapping Attacks to Impact and Mitigations

6. Association of Attacks to Use Cases

Different attacks can have different impact and/or mitigation depending on the use case, so we would like to make this association in our analysis. However since there is a potentially unbounded list of use cases, we categorize the attacks with respect to the common themes of the use cases as identified in the Use Case Common Themes section of the DetNet Use Cases draft [RFC8578].

See also Figure 2 for a mapping of the impact of attacks per use case by industry.

6.1. Use Cases by Common Themes

In this section we review each theme and discuss the attacks that are applicable to that theme, as well as anything specific about the impact and mitigations for that attack with respect to that theme. The table Figure 5 then provides a summary of the attacks that are applicable to each theme.

6.1.1. Network Layer – AVB/TSN Ethernet

DetNet is expected to run over various transmission mediums, with Ethernet being explicitly supported. Attacks such as Delay or Reconnaissance might be implemented differently on a different transmission medium, however the impact on the DetNet as a whole would be essentially the same. We thus conclude that all attacks and impacts that would be applicable to DetNet over Ethernet (i.e. all those named in this draft) would also be applicable to DetNet over other transmission mediums.

With respect to mitigations, some methods are specific to the Ethernet medium, for example time-aware scheduling using 802.1Qbv can protect against excessive use of bandwidth at the ingress - for other mediums, other mitigations would have to be implemented to provide analogous protection.

6.1.2. Central Administration

A DetNet network is expected to be controlled by a centralized network configuration and control system (CNC). Such a system may be in a single central location, or it may be distributed across multiple control entities that function together as a unified control system for the network.

In this draft we distinguish between attacks on the DetNet Control plane vs. Data plane. But is an attack affecting control plane packets synonymous with an attack on the CNC itself? For purposes of this draft let us consider an attack on the CNC itself to be out of scope, and consider all attacks named in this draft which are relevant to control plane packets to be relevant to this theme, including Path Manipulation, Path Choice, Control Packet Modification or Injection, Reconnaissance and Attacks on Time Sync Mechanisms.

6.1.3. Hot Swap

A DetNet network is not expected to be "plug and play" - it is expected that there is some centralized network configuration and control system. However, the ability to "hot swap" components (e.g. due to malfunction) is similar enough to "plug and play" that this kind of behavior may be expected in DetNet networks, depending on the implementation.

An attack surface related to Hot Swap is that the DetNet network must at least consider input at runtime from devices that were not part of the initial configuration of the network. Even a "perfect" (or "hitless") replacement of a device at runtime would not necessarily be ideal, since presumably one would want to distinguish it from the original for OAM purposes (e.g. to report hot swap of a failed device).

This implies that an attack such as Flow Modification, Spoofing or Inter-segment (which could introduce packets from a "new" device (i.e. one heretofore unknown on the network) could be used to exploit the need to consider such packets (as opposed to rejecting them out of hand as one would do if one did not have to consider introduction of a new device).

Similarly if the network was designed to support runtime replacement of a clock device, then presence (or apparent presence) and thus consideration of packets from a new such device could affect the network, or the time sync of the network, for example by initiating a new Best Master Clock selection process. Thus attacks on time sync should be considered when designing hot swap type functionality.

6.1.4. Data Flow Information Models

Data Flow Information Models specific to DetNet networks are to be specified by DetNet. Thus they are "new" and thus potentially present a new attack surface. Does the threat take advantage of any aspect of our new Data Flow Info Models?

This is TBD, thus there are no specific entries in our table, however that does not imply that there could be no relevant attacks.

6.1.5. L2 and L3 Integration

A DetNet network integrates Layer 2 (bridged) networks (e.g. AVB/TSN LAN) and Layer 3 (routed) networks via the use of well-known protocols such as IPv6, MPLS-PW, and Ethernet. Presumably security considerations applicable directly to those individual protocols is not specific to DetNet, and thus out of scope for this draft. However enabling DetNet to coordinate Layer 2 and Layer 3 behavior will require some additions to existing protocols (see draft-dt-detnet-dp-alt) and any such new work can introduce new attack surfaces.

This is TBD, thus there are no specific entries in our table, however that does not imply that there could be no relevant attacks.

6.1.6. End-to-End Delivery

Packets sent over DetNet are guaranteed not to be dropped by the network due to congestion. (Packets may however be dropped for intended reasons, e.g. per security measures).

A Data plane attack may force packets to be dropped, for example a "long" Delay or Replication/Elimination or Flow Modification attack.

The same result might be obtained by a Control plane attack, e.g. Path Manipulation or Signaling Packet Modification.

It may be that such attacks are limited to Internal MITM attackers, but other possibilities should be considered.

An attack may also cause packets that should not be delivered to be delivered, such as by forcing packets from one (e.g. replicated) path to be preferred over another path when they should not be (Replication attack), or by Flow Modification, or by Path Choice or Packet Injection. A Time Sync attack could cause a system that was expecting certain packets at certain times to accept unintended packets based on compromised system time or time windowing in the scheduler.

6.1.7. Proprietary Deterministic Ethernet Networks

There are many proprietary non-interoperable deterministic Ethernet-based networks currently available; DetNet is intended to provide an open-standards-based alternative to such networks. In cases where a DetNet intersects with remnants of such networks or their protocols, such as by protocol emulation or access to such a network via a gateway, new attack surfaces can be opened.

For example an Inter-Segment or Control plane attack such as Path Manipulation, Path Choice or Control Packet Modification/Injection could be used to exploit commands specific to such a protocol, or that are interpreted differently by the different protocols or gateway.

6.1.8. Replacement for Proprietary Fieldbuses

There are many proprietary "field buses" used in today's industrial and other industries; DetNet is intended to provide an open-standards-based alternative to such buses. In cases where a DetNet intersects with such fieldbuses or their protocols, such as by protocol emulation or access via a gateway, new attack surfaces can be opened.

For example an Inter-Segment or Control plane attack such as Path Manipulation, Path Choice or Control Packet Modification/Injection could be used to exploit commands specific to such a protocol, or that are interpreted differently by the different protocols or gateway.

6.1.9. Deterministic vs Best-Effort Traffic

DetNet is intended to support coexistence of time-sensitive operational (OT, deterministic) traffic and information (IT, "best effort") traffic on the same ("unified") network.

The presence of IT traffic on a network carrying OT traffic has long been considered insecure design [reference needed here]. With DetNet, this coexistence will become more common, and mitigations

will need to be established. The fact that the IT traffic on a DetNet is limited to a corporate controlled network makes this a less difficult problem compared to being exposed to the open Internet, however this aspect of DetNet security should not be underestimated.

Most of the themes described in this draft address OT (reserved) streams - this item is intended to address issues related to IT traffic on a DetNet.

An Inter-segment attack can flood the network with IT-type traffic with the intent of disrupting handling of IT traffic, and/or the goal of interfering with OT traffic. Presumably if the stream reservation and isolation of the DetNet is well-designed (better-designed than the attack) then interference with OT traffic should not result from an attack that floods the network with IT traffic.

However the DetNet's handling of IT traffic may not (by design) be as resilient to DOS attack, and thus designers must be otherwise prepared to mitigate DOS attacks on IT traffic in a DetNet.

6.1.10. Deterministic Flows

Reserved bandwidth data flows (deterministic flows) must provide the allocated bandwidth, and must be isolated from each other.

A Spoofing or Inter-segment attack which adds packet traffic to a bandwidth-reserved stream could cause that stream to occupy more bandwidth than it is allocated, resulting in interference with other deterministic flows.

A Flow Modification or Spoofing or Header Manipulation or Control Packet Modification attack could cause packets from one flow to be directed to another flow, thus breaching isolation between the flows.

6.1.11. Unused Reserved Bandwidth

If bandwidth reservations are made for a stream but the associated bandwidth is not used at any point in time, that bandwidth is made available on the network for best-effort traffic. If the owner of the reserved stream then starts transmitting again, the bandwidth is no longer available for best-effort traffic, on a moment-to-moment basis. (Such "temporarily available" bandwidth is not available for time-sensitive traffic, which must have its own reservation).

An Inter-segment attack could flood the network with IT traffic, interfering with the intended IT traffic.

A Flow Modification or Spoofing or Control Packet Modification or Injection attack could cause extra bandwidth to be reserved by a new or existing stream, thus making it unavailable for use by best-effort traffic.

6.1.12. Interoperability

The DetNet network specifications are intended to enable an ecosystem in which multiple vendors can create interoperable products, thus promoting device diversity and potentially higher numbers of each device manufactured. Does the threat take advantage of differences in implementation of "interoperable" products made by different vendors?

This is TBD, thus there are no specific entries in our table, however that does not imply that there could be no relevant attacks.

6.1.13. Cost Reductions

The DetNet network specifications are intended to enable an ecosystem in which multiple vendors can create interoperable products, thus promoting higher numbers of each device manufactured, promoting cost reduction and cost competition among vendors. Does the threat take advantage of "low cost" HW or SW components or other "cost-related shortcuts" that might be present in devices?

This is TBD, thus there are no specific entries in our table, however that does not imply that there could be no relevant attacks.

6.1.14. Insufficiently Secure Devices

The DetNet network specifications are intended to enable an ecosystem in which multiple vendors can create interoperable products, thus promoting device diversity and potentially higher numbers of each device manufactured. Does the threat attack "naivete" of SW, for example SW that was not designed to be sufficiently secure (or secure at all) but is deployed on a DetNet network that is intended to be highly secure? (For example IoT exploits like the Mirai video-camera botnet ([MIRAI])).

This is TBD, thus there are no specific entries in our table, however that does not imply that there could be no relevant attacks.

6.1.15. DetNet Network Size

DetNet networks range in size from very small, e.g. inside a single industrial machine, to very large, for example a Utility Grid network spanning a whole country.

The size of the network might be related to how the attack is introduced into the network, for example if the entire network is local, there is a threat that power can be cut to the entire network. If the network is large, perhaps only a part of the network is attacked.

A Delay attack might be as relevant to a small network as to a large network, although the amount of delay might be different.

Attacks sourced from IT traffic might be more likely in large networks, since more people might have access to the network. Similarly Path Manipulation, Path Choice and Time Sync attacks seem more likely relevant to large networks.

6.1.16. Multiple Hops

Large DetNet networks (e.g. a Utility Grid network) may involve many "hops" over various kinds of links for example radio repeaters, microwave links, fiber optic links, etc..

An attack that takes advantage of flaws (or even normal operation) in the device drivers for the various links (through internal knowledge of how the individual driver or firmware operates, perhaps like the Stuxnet attack) could take proportionately greater advantage of this topology. We don't currently have an attack like this defined; we have only "protocol" (time or packet) based attacks. Perhaps we need to define an attack like this? Or is that out of scope for DetNet?

It is also possible that this DetNet topology will not be in as common use as other more homogeneous topologies so there may be more opportunity for attackers to exploit software and/or protocol flaws in the implementations which have not been wrung out by extensive use, particularly in the case of early adopters.

Of the attacks we have defined, the ones identified above as relevant to "large" networks seem to be most relevant.

6.1.17. Level of Service

A DetNet is expected to provide means to configure the network that include querying network path latency, requesting bounded latency for a given stream, requesting worst case maximum and/or minimum latency for a given path or stream, and so on. It is an expected case that the network cannot provide a given requested service level. In such cases the network control system should reply that the requested service level is not available (as opposed to accepting the parameter but then not delivering the desired behavior).

Control plane attacks such as Signaling Packet Modification and Injection could be used to modify or create control traffic that could interfere with the process of a user requesting a level of service and/or the network's reply.

Reconnaissance could be used to characterize flows and perhaps target specific flows for attack via the Control plane as noted above.

6.1.18. Bounded Latency

DetNet provides the expectation of guaranteed bounded latency.

Delay attacks can cause packets to miss their agreed-upon latency boundaries.

Time Sync attacks can corrupt the system's time reference, resulting in missed latency deadlines (with respect to the "correct" time reference).

6.1.19. Low Latency

Applications may require "extremely low latency" however depending on the application these may mean very different latency values; for example "low latency" across a Utility grid network is on a different time scale than "low latency" in a motor control loop in a small machine. The intent is that the mechanisms for specifying desired latency include wide ranges, and that architecturally there is nothing to prevent arbitrarily low latencies from being implemented in a given network.

Attacks on the Control plane (as described in the Level of Service theme) and Delay and Time attacks (as described in the Bounded Latency theme) both apply here.

6.1.20. Bounded Jitter (Latency Variation)

DetNet is expected to provide bounded jitter (packet to packet latency variation).

Delay attacks can cause packets to vary in their arrival times, resulting in packet to packet latency variation, thereby violating the jitter specification.

6.1.21. Symmetrical Path Delays

Some applications would like to specify that the transit delay time values be equal for both the transmit and return paths.

Delay attacks can cause path delays to differ.

Time Sync attacks can corrupt the system's time reference, resulting in differing path delays (with respect to the "correct" time reference).

6.1.22. Reliability and Availability

DetNet based systems are expected to be implemented with essentially arbitrarily high availability (for example 99.9999% up time, or even 12 nines). The intent is that the DetNet designs should not make any assumptions about the level of reliability and availability that may be required of a given system, and should define parameters for communicating these kinds of metrics within the network.

Any attack on the system, of any type, can affect its overall reliability and availability, thus in our table we have marked every attack. Since every DetNet depends to a greater or lesser degree on reliability and availability, this essentially means that all networks have to mitigate all attacks, which to a greater or lesser degree defeats the purpose of associating attacks with use cases. It also underscores the difficulty of designing "extremely high reliability" networks. I hope that in future drafts we can say something more useful here.

6.1.23. Redundant Paths

DetNet based systems are expected to be implemented with essentially arbitrarily high reliability/availability. A strategy used by DetNet for providing such extraordinarily high levels of reliability is to provide redundant paths that can be seamlessly switched between, all the while maintaining the required performance of that system.

Replication-related attacks are by definition applicable here. Control plane attacks can also interfere with the configuration of redundant paths.

6.1.24. Security Measures

A DetNet network must be made secure against devices failures, attackers, misbehaving devices, and so on. Does the threat affect such security measures themselves, e.g. by attacking SW designed to protect against device failure?

This is TBD, thus there are no specific entries in our table, however that does not imply that there could be no relevant attacks.

6.2. Attack Types by Use Case Common Theme

The following table lists the attacks of Section 3, assigning a number to each type of attack. That number is then used as a short form identifier for the attack in Figure 5.

Attack	Section
1 Delay Attack	Section 3.2.1
2 DetNet Flow Modification or Spoofing	Section 3.2.2
3 Inter-Segment Attack	Section 3.2.3
4 Replication: Increased attack surface	Section 3.2.4.1
5 Replication-related Header Manipulation	Section 3.2.4.2
6 Path Manipulation	Section 3.2.5.1
7 Path Choice: Increased Attack Surface	Section 3.2.5.2
8 Control or Signaling Packet Modification	Section 3.2.6.1
9 Control or Signaling Packet Injection	Section 3.2.6.2
10 Reconnaissance	Section 3.2.7
11 Attacks on Time Sync Mechanisms	Section 3.2.8

Figure 4: List of Attacks

The following table maps the use case themes presented in this memo to the attacks of Figure 4. Each row specifies a theme, and the attacks relevant to this theme are marked with a '+'.

Theme	Attack										
	1	2	3	4	5	6	7	8	9	10	11
Network Layer - AVB/TSN Eth.	+	+	+	+	+	+	+	+	+	+	+
Central Administration						+	+	+	+	+	+

Hot Swap		+	+									+
Data Flow Information Models												
L2 and L3 Integration												
End-to-end Delivery	+	+	+	+	+	+	+	+	+	+		+
Proprietary Deterministic Ethernet Networks			+			+	+	+	+			
Replacement for Proprietary Fieldbuses			+			+	+	+	+			
Deterministic vs. Best-Effort Traffic			+									
Deterministic Flows		+	+		+	+		+				
Unused Reserved Bandwidth		+	+					+	+			
Interoperability												
Cost Reductions												
Insufficiently Secure Devices												
DetNet Network Size	+					+	+					+
Multiple Hops	+	+				+	+					+
Level of Service								+	+	+		
Bounded Latency	+											+
Low Latency	+							+	+	+	+	+
Bounded Jitter	+											
Symmetric Path Delays	+											+
Reliability and Availability	+	+	+	+	+	+	+	+	+	+	+	+
Redundant Paths				+	+			+	+			
Security Measures												

+-----+--+--+--+--+--+--+--+--+--+--+--+--+--+--+

Figure 5: Mapping Between Themes and Attacks

6.3. Security Considerations for OAM Traffic

This section considers DetNet-specific security considerations for packet traffic that is generated and transmitted over a DetNet as part of OAM (Operations, Administration and Maintenance). For purposes of this discussion, OAM traffic falls into one of two basic types:

- o OAM traffic generated by the network itself. The additional bandwidth required for such packets is added by the network administration, presumably transparent to the customer. Security considerations for such traffic are not DetNet-specific (apart from such traffic being subject to the same DetNet-specific security considerations as any other DetNet data flow) and are thus not covered in this document.
- o OAM traffic generated by the customer. From a DetNet security point of view, DetNet security considerations for such traffic are exactly the same as for any other customer data flows.

Thus OAM traffic presents no additional (i.e. OAM-specific) DetNet security considerations.

7. DetNet Technology-Specific Threats

Section 3 described threats which are independent of the DetNet implementation. This section considers threats related to the specific technologies referenced in [I-D.ietf-detnet-data-plane-framework] which have not already been enumerated in Section 3.

As in this document in general, this section only enumerates security aspects which are unique to providing the specific quality of service aspects of DetNet, which are primarily to deliver data flows with extremely low packet loss rates and bounded end-to-end delivery latency. The primary considerations for the data plane specifically are to maintain integrity of data and delivery of the associated DetNet service traversing the DetNet network.

As noted in [RFC8655], DetNet operates at the IP layer ([I-D.ietf-detnet-ip]) and delivers service over sub-layer technologies such as MPLS ([I-D.ietf-detnet-mpls]) and IEEE 802.1 Time-Sensitive Networking (TSN) ([I-D.ietf-detnet-ip-over-tsn]).

Application flows can be protected through whatever means is provided by the underlying technology. For example, technology-specific encryption may be used, such as that provided by IPSec [RFC4301] for IP flows and/or by an underlying sub-net using MACSec [IEEE802.1AE-2018] for IP over Ethernet (Layer-2) flows.

Sections below discuss threats specific to IP, MPLS, and TSN in more detail.

7.1. IP

The IP protocol has a long history of security considerations and architectural protection mechanisms. From a data plane perspective DetNet does not add or modify any IP header information, and its use as a DetNet Data Plane does not introduce any new security issues that were not there before, apart from those already described in the data-plane-independent threats section Section 3.

Thus the security considerations for a DetNet based on an IP data plane are purely inherited from the rich IP Security literature and code/application base, and the data-plane-independent section of this document.

7.2. MPLS

An MPLS network carrying DetNet traffic is expected to be a "well-managed" network. Given that this is the case, it is difficult for an attacker to pass a raw MPLS encoded packet into a network because operators have considerable experience at excluding such packets at the network boundaries, as well as excluding MPLS packets being inserted through the use of a tunnel.

MPLS security is discussed extensively in [RFC5920] ("Security Framework for MPLS and GMPLS Networks") to which the reader is referred.

[RFC6941] builds on [RFC5920] by providing additional security considerations that are applicable to the MPLS-TP extensions appropriate to the MPLS Transport Profile [RFC5921], and thus to the operation of DetNet over some types of MPLS network.

[RFC5921] introduces to MPLS new Operations, Administration, and Maintenance (OAM) capabilities, a transport-oriented path protection mechanism, and strong emphasis on static provisioning supported by network management systems.

The operation of DetNet over an MPLS network is modeled on the operation of multi-segment pseudowires (MS-PW). Thus for guidance on

securing the DetNet elements of DetNet over MPLS the reader is referred to the MS-PW security mechanisms as defined in [RFC4447], [RFC3931], [RFC3985], [RFC6073], and [RFC6478].

Having attended to the conventional aspects of network security it is necessary to attend to the dynamic aspects. The closest experience that the IETF has with securing protocols that are sensitive to manipulation of delay are the two way time transfer protocols (TWTT), which are NTP [RFC5905] and Precision Time Protocol [IEEE1588]. The security requirements for these are described in [RFC7384].

One particular problem that has been observed in operational tests of TWTT protocols is the ability for two closely but not completely synchronized streams to beat and cause a sudden phase hit to one of the streams. This can be mitigated by the careful use of a scheduling system in the underlying packet transport.

Further consideration of protection against dynamic attacks is work in progress.

7.3. TSN

Editor's Note: To Be Written.

8. Appendix A: DetNet Draft Security-Related Statements

This section collects the various statements in the currently existing DetNet Working Group drafts. For each draft, the section name and number of the quoted section is shown. The text shown here is the work of the original draft authors, quoted verbatim from the drafts. The intention is to explicitly quote all relevant text, not to summarize it.

8.1. Architecture (draft 8)

8.1.1. Fault Mitigation (sec 4.5)

One key to building robust real-time systems is to reduce the infinite variety of possible failures to a number that can be analyzed with reasonable confidence. DetNet aids in the process by providing filters and policers to detect DetNet packets received on the wrong interface, or at the wrong time, or in too great a volume, and to then take actions such as discarding the offending packet, shutting down the offending DetNet flow, or shutting down the offending interface.

It is also essential that filters and service remarking be employed at the network edge to prevent non-DetNet packets from being mistaken

for DetNet packets, and thus impinging on the resources allocated to DetNet packets.

There exist techniques, at present and/or in various stages of standardization, that can perform these fault mitigation tasks that deliver a high probability that misbehaving systems will have zero impact on well-behaved DetNet flows, except of course, for the receiving interface(s) immediately downstream of the misbehaving device. Examples of such techniques include traffic policing functions (e.g. [RFC2475]) and separating flows into per-flow rate-limited queues.

8.1.2. Security Considerations (sec 7)

Security in the context of Deterministic Networking has an added dimension; the time of delivery of a packet can be just as important as the contents of the packet, itself. A man-in-the-middle attack, for example, can impose, and then systematically adjust, additional delays into a link, and thus disrupt or subvert a real-time application without having to crack any encryption methods employed. See [RFC7384] for an exploration of this issue in a related context.

Furthermore, in a control system where millions of dollars of equipment, or even human lives, can be lost if the DetNet QoS is not delivered, one must consider not only simple equipment failures, where the box or wire instantly becomes perfectly silent, but bizarre errors such as can be caused by software failures. Because there is essential no limit to the kinds of failures that can occur, protecting against realistic equipment failures is indistinguishable, in most cases, from protecting against malicious behavior, whether accidental or intentional.

Security must cover:

- o Protection of the signaling protocol
- o Authentication and authorization of the controlling nodes
- o Identification and shaping of the flows

8.2. Data Plane Alternatives (draft 4)

8.2.1. Security Considerations (sec 7)

This document does not add any new security considerations beyond what the referenced technologies already have.

8.3. Problem Statement (draft 5)

8.3.1. Security Considerations (sec 5)

Security in the context of Deterministic Networking has an added dimension; the time of delivery of a packet can be just as important as the contents of the packet, itself. A man-in-the-middle attack, for example, can impose, and then systematically adjust, additional delays into a link, and thus disrupt or subvert a real-time application without having to crack any encryption methods employed. See [RFC7384] for an exploration of this issue in a related context.

Typical control networks today rely on complete physical isolation to prevent rogue access to network resources. DetNet enables the virtualization of those networks over a converged IT/OT infrastructure. Doing so, DetNet introduces an additional risk that flows interact and interfere with one another as they share physical resources such as Ethernet trunks and radio spectrum. The requirement is that there is no possible data leak from and into a deterministic flow, and in a more general fashion there is no possible influence whatsoever from the outside on a deterministic flow. The expectation is that physical resources are effectively associated with a given flow at a given point of time. In that model, Time Sharing of physical resources becomes transparent to the individual flows which have no clue whether the resources are used by other flows at other times.

Security must cover:

- o Protection of the signaling protocol
- o Authentication and authorization of the controlling nodes
- o Identification and shaping of the flows
- o Isolation of flows from leakage and other influences from any activity sharing physical resources

8.4. Use Cases (draft 11)

8.4.1. (Utility Networks) Security Current Practices and Limitations (sec 3.2.1)

Grid monitoring and control devices are already targets for cyber attacks, and legacy telecommunications protocols have many intrinsic network-related vulnerabilities. For example, DNP3, Modbus, PROFIBUS/PROFINET, and other protocols are designed around a common paradigm of request and respond. Each protocol is designed for a

master device such as an HMI (Human Machine Interface) system to send commands to subordinate slave devices to retrieve data (reading inputs) or control (writing to outputs). Because many of these protocols lack authentication, encryption, or other basic security measures, they are prone to network-based attacks, allowing a malicious actor or attacker to utilize the request-and-respond system as a mechanism for command-and-control like functionality. Specific security concerns common to most industrial control, including utility telecommunication protocols include the following:

- o Network or transport errors (e.g. malformed packets or excessive latency) can cause protocol failure.
- o Protocol commands may be available that are capable of forcing slave devices into inoperable states, including powering-off devices, forcing them into a listen-only state, disabling alarming.
- o Protocol commands may be available that are capable of restarting communications and otherwise interrupting processes.
- o Protocol commands may be available that are capable of clearing, erasing, or resetting diagnostic information such as counters and diagnostic registers.
- o Protocol commands may be available that are capable of requesting sensitive information about the controllers, their configurations, or other need-to-know information.
- o Most protocols are application layer protocols transported over TCP; therefore it is easy to transport commands over non-standard ports or inject commands into authorized traffic flows.
- o Protocol commands may be available that are capable of broadcasting messages to many devices at once (i.e. a potential DoS).
- o Protocol commands may be available to query the device network to obtain defined points and their values (i.e. a configuration scan).
- o Protocol commands may be available that will list all available function codes (i.e. a function scan).
- o These inherent vulnerabilities, along with increasing connectivity between IT and OT networks, make network-based attacks very feasible.

- o Simple injection of malicious protocol commands provides control over the target process. Altering legitimate protocol traffic can also alter information about a process and disrupt the legitimate controls that are in place over that process. A man-in-the-middle attack could provide both control over a process and misrepresentation of data back to operator consoles.

8.4.2. (Utility Networks) Security Trends in Utility Networks (sec 3.3.3)

Although advanced telecommunications networks can assist in transforming the energy industry by playing a critical role in maintaining high levels of reliability, performance, and manageability, they also introduce the need for an integrated security infrastructure. Many of the technologies being deployed to support smart grid projects such as smart meters and sensors can increase the vulnerability of the grid to attack. Top security concerns for utilities migrating to an intelligent smart grid telecommunications platform center on the following trends:

- o Integration of distributed energy resources
- o Proliferation of digital devices to enable management, automation, protection, and control
- o Regulatory mandates to comply with standards for critical infrastructure protection
- o Migration to new systems for outage management, distribution automation, condition-based maintenance, load forecasting, and smart metering
- o Demand for new levels of customer service and energy management

This development of a diverse set of networks to support the integration of microgrids, open-access energy competition, and the use of network-controlled devices is driving the need for a converged security infrastructure for all participants in the smart grid, including utilities, energy service providers, large commercial and industrial, as well as residential customers. Securing the assets of electric power delivery systems (from the control center to the substation, to the feeders and down to customer meters) requires an end-to-end security infrastructure that protects the myriad of telecommunications assets used to operate, monitor, and control power flow and measurement.

"Cyber security" refers to all the security issues in automation and telecommunications that affect any functions related to the operation

of the electric power systems. Specifically, it involves the concepts of:

- o Integrity : data cannot be altered undetectably
- o Authenticity : the telecommunications parties involved must be validated as genuine
- o Authorization : only requests and commands from the authorized users can be accepted by the system
- o Confidentiality : data must not be accessible to any unauthenticated users

When designing and deploying new smart grid devices and telecommunications systems, it is imperative to understand the various impacts of these new components under a variety of attack situations on the power grid. Consequences of a cyber attack on the grid telecommunications network can be catastrophic. This is why security for smart grid is not just an ad hoc feature or product, it's a complete framework integrating both physical and Cyber security requirements and covering the entire smart grid networks from generation to distribution. Security has therefore become one of the main foundations of the utility telecom network architecture and must be considered at every layer with a defense-in-depth approach. Migrating to IP based protocols is key to address these challenges for two reasons:

- o IP enables a rich set of features and capabilities to enhance the security posture
- o IP is based on open standards, which allows interoperability between different vendors and products, driving down the costs associated with implementing security solutions in OT networks.

Securing OT (Operation technology) telecommunications over packet-switched IP networks follow the same principles that are foundational for securing the IT infrastructure, i.e., consideration must be given to enforcing electronic access control for both person-to-machine and machine-to-machine communications, and providing the appropriate levels of data privacy, device and platform integrity, and threat detection and mitigation.

Existing power automation security standards can inform network security. For example the NERC CIP (North American Electric Reliability Corporation Critical Infrastructure Protection) plan is a set of requirements designed to secure the assets required for operating North America's bulk electric system. Another standardized

security control technique is Segmentation (zones and conduits including access control).

The requirements in Industrial Automation and Control Systems (IACS) are quite similar, especially in new scenarios such as Industry 4.0/ Digital Factory where workflows and protocols cross zones, segments, and entities. IEC 62443 (ISA99) defines security for IACS, typically for installations in other critical infrastructure such as oil and gas.

Availability and integrity are the most important security objectives for the lower layers of such networks; confidentiality and privacy are relevant if customer or market data is involved, typically handled by higher layers.

8.4.3. (BAS) Security Considerations (sec 4.2.4)

When BAS field networks were developed it was assumed that the field networks would always be physically isolated from external networks and therefore security was not a concern. In today's world many BASs are managed remotely and are thus connected to shared IP networks and so security is definitely a concern, yet security features are not available in the majority of BAS field network deployments .

The management network, being an IP-based network, has the protocols available to enable network security, but in practice many BAS systems do not implement even the available security features such as device authentication or encryption for data in transit.

8.4.4. (6TiSCH) Security Considerations (sec 5.3.3)

On top of the classical requirements for protection of control signaling, it must be noted that 6TiSCH networks operate on limited resources that can be depleted rapidly in a DoS attack on the system, for instance by placing a rogue device in the network, or by obtaining management control and setting up unexpected additional paths.

8.4.5. (Cellular radio) Security Considerations (sec 6.1.5)

Establishing time-sensitive streams in the network entails reserving networking resources for long periods of time. It is important that these reservation requests be authenticated to prevent malicious reservation attempts from hostile nodes (or accidental misconfiguration). This is particularly important in the case where the reservation requests span administrative domains. Furthermore, the reservation information itself should be digitally signed to

reduce the risk of a legitimate node pushing a stale or hostile configuration into another networking node.

Note: This is considered important for the security policy of the network, but does not affect the core DetNet architecture and design.

8.4.6. (Industrial M2M) Communication Today (sec 7.2)

Industrial network scenarios require advanced security solutions. Many of the current industrial production networks are physically separated. Preventing critical flows from be leaked outside a domain is handled today by filtering policies that are typically enforced in firewalls.

9. IANA Considerations

This memo includes no requests from IANA.

10. Security Considerations

The security considerations of DetNet networks are presented throughout this document.

11. Informative References

[ARINC664P7]

ARINC, "ARINC 664 Aircraft Data Network, Part 7, Avionics Full-Duplex Switched Ethernet Network", 2009.

[I-D.ietf-detnet-data-plane-framework]

Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane Framework", draft-ietf-detnet-data-plane-framework-02 (work in progress), September 2019.

[I-D.ietf-detnet-ip]

Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: IP", draft-ietf-detnet-ip-03 (work in progress), October 2019.

[I-D.ietf-detnet-ip-over-tsn]

Varga, B., Farkas, J., Malis, A., and S. Bryant, "DetNet Data Plane: IP over IEEE 802.1 Time Sensitive Networking (TSN)", draft-ietf-detnet-ip-over-tsn-01 (work in progress), October 2019.

- [I-D.ietf-detnet-mpls]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: MPLS", draft-ietf-detnet-mpls-03 (work in progress), October 2019.
- [I-D.varga-detnet-service-model]
Varga, B. and J. Farkas, "DetNet Service Model", draft-varga-detnet-service-model-02 (work in progress), May 2017.
- [IEEE1588]
IEEE, "IEEE 1588 Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems Version 2", 2008.
- [IEEE802.1AE-2018]
IEEE Standards Association, "IEEE Std 802.1AE-2018 MAC Security (MACsec)", 2018,
<<https://ieeexplore.ieee.org/document/8585421>>.
- [IEEE802.1Qch-2017]
IEEE Standards Association, "IEEE Standard for Local and metropolitan area networks--Bridges and Bridged Networks--Amendment 29: Cyclic Queuing and Forwarding", 2017,
<<https://ieeexplore.ieee.org/document/7961303>>.
- [MIRAI]
krebsonsecurity.com, "<https://krebsonsecurity.com/2016/10/hacked-cameras-dvrs-powered-todays-massive-internet-outage/>", 2016.
- [RFC2475]
Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., and W. Weiss, "An Architecture for Differentiated Services", RFC 2475, DOI 10.17487/RFC2475, December 1998,
<<https://www.rfc-editor.org/info/rfc2475>>.
- [RFC3552]
Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", BCP 72, RFC 3552, DOI 10.17487/RFC3552, July 2003,
<<https://www.rfc-editor.org/info/rfc3552>>.
- [RFC3931]
Lau, J., Ed., Townsley, M., Ed., and I. Goyret, Ed., "Layer Two Tunneling Protocol - Version 3 (L2TPv3)", RFC 3931, DOI 10.17487/RFC3931, March 2005,
<<https://www.rfc-editor.org/info/rfc3931>>.

- [RFC3985] Bryant, S., Ed. and P. Pate, Ed., "Pseudo Wire Emulation Edge-to-Edge (PWE3) Architecture", RFC 3985, DOI 10.17487/RFC3985, March 2005, <<https://www.rfc-editor.org/info/rfc3985>>.
- [RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <<https://www.rfc-editor.org/info/rfc4301>>.
- [RFC4447] Martini, L., Ed., Rosen, E., El-Aawar, N., Smith, T., and G. Heron, "Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)", RFC 4447, DOI 10.17487/RFC4447, April 2006, <<https://www.rfc-editor.org/info/rfc4447>>.
- [RFC5905] Mills, D., Martin, J., Ed., Burbank, J., and W. Kasch, "Network Time Protocol Version 4: Protocol and Algorithms Specification", RFC 5905, DOI 10.17487/RFC5905, June 2010, <<https://www.rfc-editor.org/info/rfc5905>>.
- [RFC5920] Fang, L., Ed., "Security Framework for MPLS and GMPLS Networks", RFC 5920, DOI 10.17487/RFC5920, July 2010, <<https://www.rfc-editor.org/info/rfc5920>>.
- [RFC5921] Bocci, M., Ed., Bryant, S., Ed., Frost, D., Ed., Levrau, L., and L. Berger, "A Framework for MPLS in Transport Networks", RFC 5921, DOI 10.17487/RFC5921, July 2010, <<https://www.rfc-editor.org/info/rfc5921>>.
- [RFC6073] Martini, L., Metz, C., Nadeau, T., Bocci, M., and M. Aissaoui, "Segmented Pseudowire", RFC 6073, DOI 10.17487/RFC6073, January 2011, <<https://www.rfc-editor.org/info/rfc6073>>.
- [RFC6478] Martini, L., Swallow, G., Heron, G., and M. Bocci, "Pseudowire Status for Static Pseudowires", RFC 6478, DOI 10.17487/RFC6478, May 2012, <<https://www.rfc-editor.org/info/rfc6478>>.
- [RFC6941] Fang, L., Ed., Niven-Jenkins, B., Ed., Mansfield, S., Ed., and R. Graveman, Ed., "MPLS Transport Profile (MPLS-TP) Security Framework", RFC 6941, DOI 10.17487/RFC6941, April 2013, <<https://www.rfc-editor.org/info/rfc6941>>.
- [RFC7384] Mizrahi, T., "Security Requirements of Time Protocols in Packet Switched Networks", RFC 7384, DOI 10.17487/RFC7384, October 2014, <<https://www.rfc-editor.org/info/rfc7384>>.

- [RFC8578] Grossman, E., Ed., "Deterministic Networking Use Cases", RFC 8578, DOI 10.17487/RFC8578, May 2019, <<https://www.rfc-editor.org/info/rfc8578>>.
- [RFC8655] Finn, N., Thubert, P., Varga, B., and J. Farkas, "Deterministic Networking Architecture", RFC 8655, DOI 10.17487/RFC8655, October 2019, <<https://www.rfc-editor.org/info/rfc8655>>.

Authors' Addresses

Tal Mizrahi
Huawei Network.IO Innovation Lab

Email: tal.mizrahi.phd@gmail.com

Ethan Grossman (editor)
Dolby Laboratories, Inc.
1275 Market Street
San Francisco, CA 94103
USA

Phone: +1 415 645 4726
Email: ethan.grossman@dolby.com
URI: <http://www.dolby.com>

Andrew J. Hacker
MistIQ Technologies, Inc
Harrisburg, PA
USA

Email: ajhacker@mistiqttech.com
URI: <http://www.mistiqttech.com>

Subir Das
Applied Communication Sciences
150 Mount Airy Road, Basking Ridge
New Jersey, 07920
USA

Email: sdas@appcomsci.com

John Dowdell
Airbus Defence and Space
Celtic Springs
Newport NP10 8FZ
United Kingdom

Email: john.dowdell.ietf@gmail.com

Henrik Austad
SINTEF Digital
Klaebuveien 153
Trondheim 7037
Norway

Email: henrik@austad.us

Norman Finn
Huawei

Email: norman.finn@mail01.huawei.com

DetNet
Internet-Draft
Intended status: Standards Track
Expires: April 29, 2020

B. Varga, Ed.
J. Farkas
Ericsson
A. Malis
Independent
S. Bryant
Futurewei Technologies
D. Fedyk
LabN Consulting, L.L.C.
October 27, 2019

DetNet Data Plane: IEEE 802.1 Time Sensitive Networking over MPLS
draft-ietf-detnet-tsn-vpn-over-mpls-01

Abstract

This document specifies the Deterministic Networking data plane when TSN networks are interconnected over a DetNet MPLS Network.

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 April 29, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must

include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Terminology	3
2.1. Terms Used in This Document	3
2.2. Abbreviations	3
2.3. Requirements Language	4
3. IEEE 802.1 TSN Over DetNet MPLS Data Plane Scenario	4
4. DetNet MPLS Data Plane	6
4.1. Overview	6
4.2. TSN over DetNet MPLS Encapsulation	7
5. TSN over MPLS Data Plane Procedures	8
5.1. Edge Node TSN Procedures	8
5.2. Edge Node DetNet Service Proxy Procedures	9
5.3. Edge Node DetNet Service and Forwarding Sub-Layer Procedures	9
6. Controller Plane (Management and Control) Considerations	10
7. Security Considerations	11
8. IANA Considerations	11
9. Acknowledgements	11
10. References	11
10.1. Normative References	11
10.2. Informative References	12
Authors' Addresses	13

1. Introduction

The Time-Sensitive Networking Task Group (TSN TG) within IEEE 802.1 Working Group deals with deterministic services through IEEE 802 networks. Deterministic Networking (DetNet) defined by IETF is a service that can be offered by a L3 network to DetNet flows. General background and concepts of DetNet can be found in [I-D.ietf-detnet-architecture].

This document specifies the use of a DetNet MPLS network to interconnect TSN nodes/network segments. DetNet MPLS data plane is defined in [I-D.ietf-detnet-mpls].

2. Terminology

2.1. Terms Used in This Document

This document uses the terminology and concepts established in the DetNet architecture [I-D.ietf-detnet-architecture] and [I-D.ietf-detnet-data-plane-framework], and [I-D.ietf-detnet-mpls]. The reader is assumed to be familiar with these documents and their terminology.

2.2. Abbreviations

The following abbreviations are used in this document:

AC	Attachment Circuit.
CE	Customer Edge equipment.
CoS	Class of Service.
CW	Control Word.
DetNet	Deterministic Networking.
DF	DetNet Flow.
FRER	Frame Replication and Elimination for Redundancy (TSN function).
L2	Layer 2.
L2VPN	Layer 2 Virtual Private Network.
L3	Layer 3.
LSR	Label Switching Router.
MPLS	Multiprotocol Label Switching.
MPLS-TE	Multiprotocol Label Switching - Traffic Engineering.
MPLS-TP	Multiprotocol Label Switching - Transport Profile.
MS-PW	Multi-Segment PseudoWire (MS-PW).
NSP	Native Service Processing.
OAM	Operations, Administration, and Maintenance.

PE	Provider Edge.
PEF	Packet Elimination Function.
PRF	Packet Replication Function.
PREOF	Packet Replication, Elimination and Ordering Functions.
POF	Packet Ordering Function.
PSN	Packet Switched Network.
PW	PseudoWire.
QoS	Quality of Service.
S-PE	Switching Provider Edge.
T-PE	Terminating Provider Edge.
TSN	Time-Sensitive Network.

2.3. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. IEEE 802.1 TSN Over DetNet MPLS Data Plane Scenario

Figure 1 shows IEEE 802.1 TSN end stations operating over a TSN aware DetNet service running over an MPLS network. DetNet Edge Nodes sit at the boundary of a DetNet domain. They are responsible for mapping non-DetNet aware L2 traffic to DetNet services. They also support the imposition and disposition of the required DetNet encapsulation. These are functionally similar to pseudowire (PW) Terminating Provider Edge (T-PE) nodes which use MPLS-TE LSPs. In this example TSN Streams are simple applications over DetNet flows. The specific of this operation are discussed later in this document.

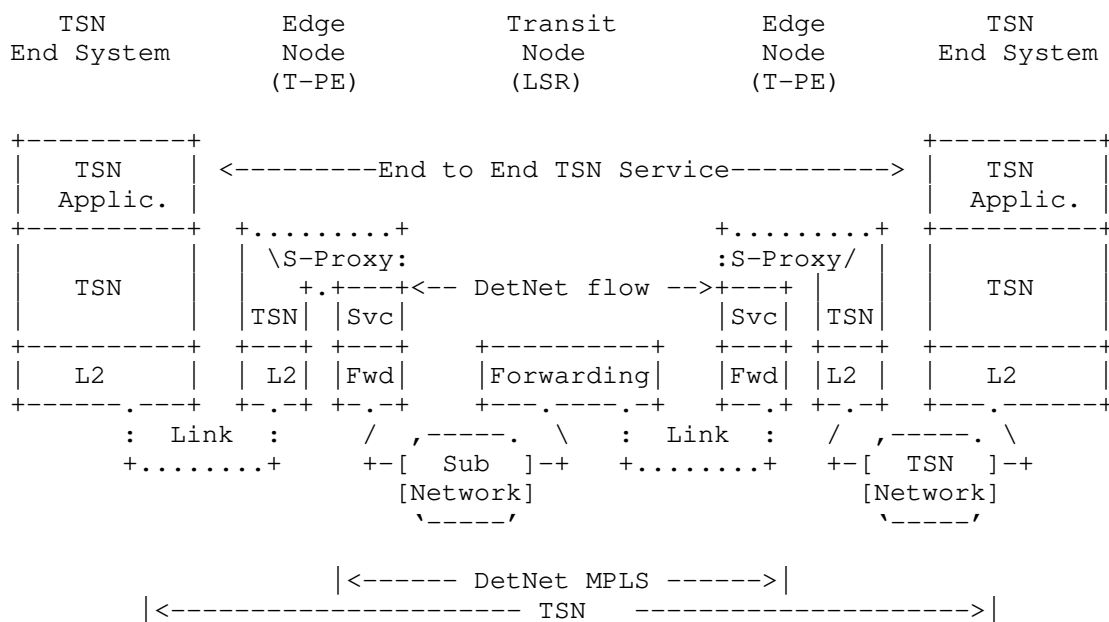


Figure 1: A TSN over DetNet MPLS Enabled Network

In this example, edge nodes provide a service proxy function that "associates" the DetNet flows and native flows (i.e., TSN Streams) at the edge of the DetNet domain. TSN streams are treated as App-flows for DetNet. The whole DetNet domain behaves as a TSN relay node for the TSN streams. The service proxy behaves as a port of that TSN relay node.

Figure 2 illustrates how DetNet can provide services for IEEE 802.1 TSN end systems, CE1 and CE2, over a DetNet enabled MPLS network. Edge nodes, E1 and E2, insert and remove required DetNet data plane encapsulation. The 'X' in the edge nodes and relay node, R1, represent a potential DetNet compound flow packet replication and elimination point. This conceptually parallels L2VPN services, and could leverage existing related solutions as discussed below.

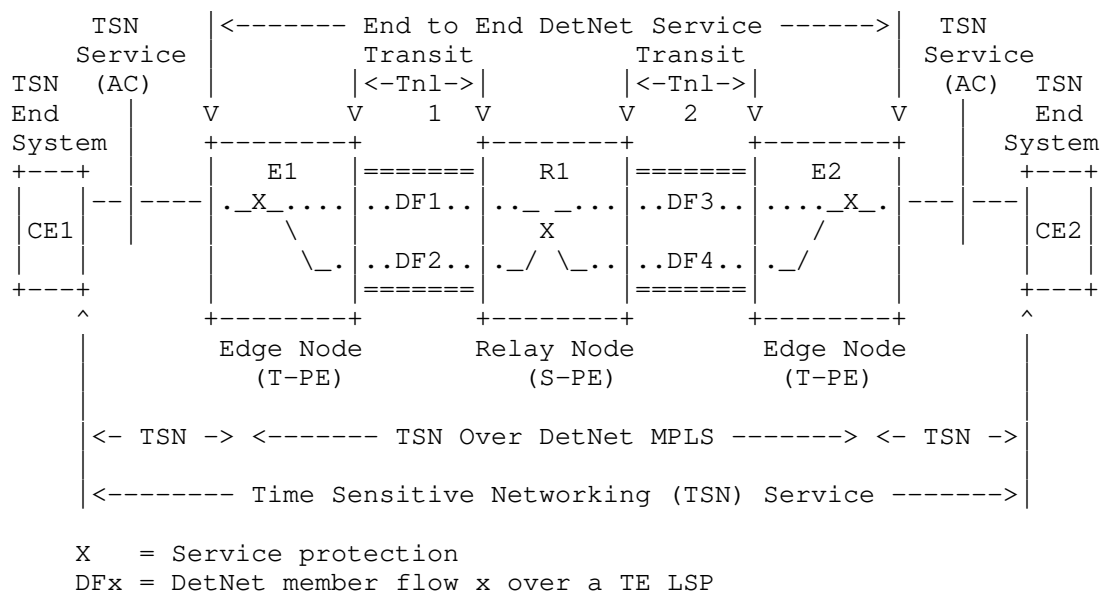


Figure 2: IEEE 802.1TSN Over DetNet

4. DetNet MPLS Data Plane

4.1. Overview

The basic approach defined in [I-D.ietf-detnet-mpls] supports the DetNet service sub-layer based on existing pseudowire (PW) encapsulations and mechanisms, and supports the DetNet forwarding sub-layer based on existing MPLS Traffic Engineering encapsulations and mechanisms.

A node operating on a DetNet flow in the Detnet service sub-layer, i.e. a node processing a DetNet packet which has the S-Label as top of stack uses the local context associated with that S-Label, for example a received F-Label, to determine what local DetNet operation(s) are applied to that packet. An S-Label may be unique when taken from the platform label space [RFC3031], which would enable correct DetNet flow identification regardless of which input interface or LSP the packet arrives on. The service sub-layer functions (i.e., PREOF) use a DetNet control word (d-CW).

The DetNet MPLS data plane builds on MPLS Traffic Engineering encapsulations and mechanisms to provide a forwarding sub-layer that is responsible for providing resource allocation and explicit routes.

The forwarding sub-layer is supported by one or more forwarding labels (F-Labels).

DetNet edge/relay nodes are DetNet service sub-layer aware, understand the particular needs of DetNet flows and provide both DetNet service and forwarding sub-layer functions. They add, remove and process d-CWs, S-Labels and F-labels as needed. MPLS enabled DetNet nodes can enhance the reliability of delivery by enabling the replication of packets where multiple copies, possibly over multiple paths, are forwarded through the DetNet domain. They can also eliminate surplus previously replicated copies of DetNet packets. MPLS (DetNet) nodes also include DetNet forwarding sub-layer functions, support for notably explicit routes, and resources allocation to eliminate (or reduce) congestion loss and jitter.

DetNet transit nodes reside wholly within a DetNet domain, and also provide DetNet forwarding sub-layer functions in accordance with the performance required by a DetNet flow carried over an LSP. Unlike other DetNet node types, transit nodes provide no service sub-layer processing.

4.2. TSN over DetNet MPLS Encapsulation

The basic encapsulation approach is to treat a TSN Stream as an app-flow from the DetNet MPLS perspective. The corresponding example shown in Figure 3.

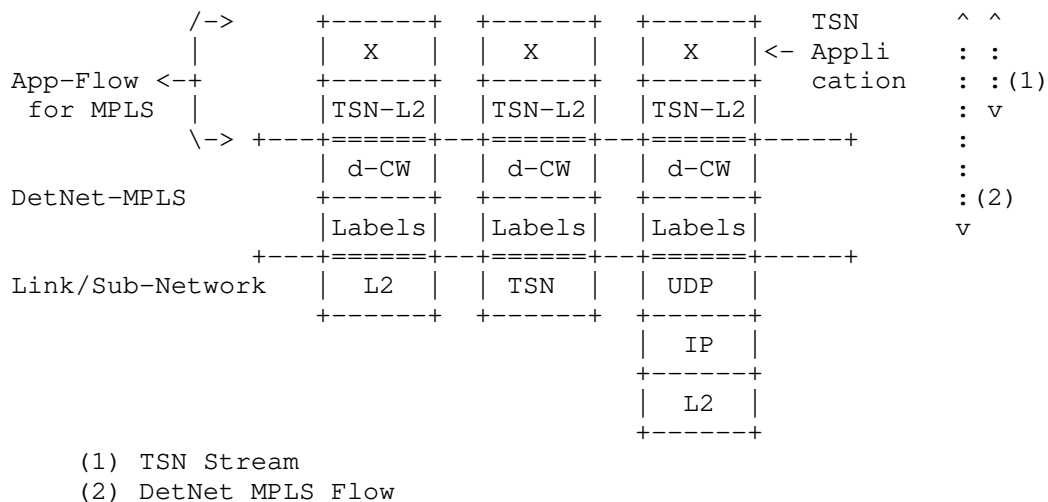


Figure 3: Example TSN over MPLS Encapsulation Formats

In the figure, "Application" indicates the application payload carried by the TSN network. "MPLS App-Flow" indicates that the TSN Stream is the payload from the perspective of the DetNet MPLS data plane defined in [I-D.ietf-detnet-mpls]. A single DetNet MPLS flow can aggregate multiple TSN Streams.

5. TSN over MPLS Data Plane Procedures

Description of Edge Nodes procedures and functions for TSN over DetNet MPLS scenario follows the concept of [RFC3985] and covers the Edge Nodes components shown on Figure 1. In this section the following procedures of DetNet Edge Nodes are described:

- o TSN related (Section 5.1)
- o DetNet Service Proxy (Section 5.2)
- o DetNet service and forwarding sub-layer (Section 5.3)

5.1. Edge Node TSN Procedures

The Time-Sensitive Networking (TSN) Task Group of the IEEE 802.1 Working Group have defined (and are defining) a number of amendments to IEEE 802.1Q [IEEE8021Q] that provide zero congestion loss and bounded latency in bridged networks. IEEE 802.1CB [IEEE8021CB] defines packet replication and elimination functions for a TSN network.

TSN specific functions are executed on the data received by the PE from the CE before presentation to the DetNet PW for transmission across the DetNet domain, or on the data received from a DetNet PW by a PE before it is output on the Attachment Circuit (AC).

TSN specific function(s) of Edge Nodes (T-PE) are belonging to the native service processing (NSP) [RFC3985] block. This is similar to other functionalities being defined by standard bodies other than the IETF (for example in case of Ethernet: stripping, overwriting or adding VLAN tags, etc.). Depending on the TSN role of the Edge Node in the end-to-end TSN service selected TSN functions must be supported.

Implementations of this document SHALL use management and control information to ensure TSN specific functions of the Edge Node according to the expectations of the connected TSN network.

5.2. Edge Node DetNet Service Proxy Procedures

The Service Proxy function maps between App-flows and DetNet flows. The DetNet Edge Node TSN function MUST support the TSN Stream identification functions and the related managed objects as defined in IEEE 802.1CB [IEEE8021CB] and IEEE P802.1CBdb [IEEEP8021CBdb] to recognize the App-flow related packets. The Service Proxy presents TSN Streams as an App-flow to a DetNet Flow.

Implementations of this document SHALL use management and control information to map a TSN Stream to a DetNet flow. N:1 mapping (aggregating multiple TSN Streams in a single DetNet flow) SHALL be supported. The management or control function that provisions flow mapping SHALL ensure that adequate resources are allocated and configured to provide proper service requirements of the mapped flows.

Due to the (intentional) similarities of the DetNet PREOF and TSN FRER functions service protection function interworking is possible between the TSN and the DetNet domains. Such service protection interworking scenarios MAY require to copy sequence number fields from TSN (L2) to PW (MPLS) encapsulation. However, such interworking is out-of-scope in this document and left for further study.

A MPLS DetNet flow is configured to carry any number of TSN flows. The DetNet flow specific bandwidth profile SHOULD match the required bandwidth of the App-flow aggregate.

5.3. Edge Node DetNet Service and Forwarding Sub-Layer Procedures

In the design of [I-D.ietf-detnet-mpls] an MPLS service label (the S-Label), similar to a pseudowire (PW) label [RFC3985], is used to identify both the DetNet flow identity and the payload MPLS payload type. The DetNet sequence number is carried in the DetNet Control word (d-CW) which carries the Data/OAM discriminator as well. In [I-D.ietf-detnet-mpls] two sequence number sizes are supported: a 16 bit sequence number and a 28 bit sequence number.

PREOF functions and the provided service recovery is available only within the DetNet domain as the DetNet flow-ID and the DetNet sequence number are not valid outside the DetNet network. MPLS (DetNet) Edge node terminates all related information elements encoded in the MPLS labels.

The LSP used to forward the DetNet packet may be of any type (MPLS-LDP, MPLS-TE, MPLS-TP [RFC5921], or MPLS-SR [I-D.ietf-spring-segment-routing-mpls]). The LSP (F-Label) label

and/or the S-Label may be used to indicate the queue processing as well as the forwarding parameters.

For further details see [I-D.ietf-detnet-mpls].

6. Controller Plane (Management and Control) Considerations

TSN Stream(s) to DetNet flow mapping related information are required only for the service proxy function of MPLS (DetNet) Edge nodes. From the Data Plane perspective there is no practical difference based on the origin of flow mapping related information (management plane or control plane).

MPLS DetNet Edge nodes are member of both the DetNet domain and the connected TSN network. From the TSN network perspective the MPLS (DetNet) Edge node has a "TSN relay node" role, so TSN specific management and control plane functionalities must be implemented. There are many similarities in the management plane techniques used in DetNet and TSN, but that is not the case for the control plane protocols. For example, RSVP-TE and MSRP behaves differently. Therefore management and control plane design is an important aspect of scenarios, where mapping between DetNet and TSN is required.

Note that, as the DetNet network is just a portion of the end to end TSN path (i.e., single hop from Ethernet perspective), some parameters (e.g., delay) may differ significantly. Since there is no interworking function the bandwidth of DetNet network is assumed to be set large enough to handle all TSN Flows it will support. At the egress of the Detnet Domain the MPLS headers are stripped and the TSN flow continues on as a normal TSN flow.

In order to use a DetNet network to interconnect TSN segments, TSN specific information must be converted to DetNet domain specific ones. TSN Stream ID(s) and stream(s) related parameters/requirements must be converted to a DetNet flow-ID and flow related parameters/requirements.

In some case it may be challenging to determine some egress node related information. For example, it may be not trivial to locate the egress point/interface of a TSN Streams with a multicast destination MAC address. Such scenarios may require interaction between control and management plane functions and between DetNet and TSN domains.

Mapping between DetNet flow identifiers and TSN Stream identifiers, if not provided explicitly, can be done by the service proxy function of an MPLS (DetNet) Edge node locally based on information provided

for configuration of the TSN Stream identification functions (e.g., Mask-and-Match Stream identification).

Triggering the setup/modification of a DetNet flow in the DetNet network is an example where management and/or control plane interactions are required between the DetNet and the TSN network.

Configuration of TSN specific functions (e.g., FRER) inside the TSN network is a TSN domain specific decision and may not be visible in the DetNet domain. Service protection interworking scenarios are left for further study.

7. Security Considerations

Security considerations for DetNet are described in detail in [I-D.ietf-detnet-security]. General security considerations are described in [I-D.ietf-detnet-architecture]. DetNet MPLS data plane specific considerations are summarized in [I-D.ietf-detnet-mpls]. The primary considerations for the data plane is to maintain integrity of data and delivery of the associated DetNet service traversing the DetNet network. Application flows can be protected through whatever means is provided by the underlying technology. For example, encryption may be used, such as that provided by IPSec [RFC4301] for IP flows and/or by an underlying sub-net using MACSec [IEEE802.1AE-2018] for IP over Ethernet (Layer-2) flows.

8. IANA Considerations

This document makes no IANA requests.

9. Acknowledgements

The authors wish to thank Norman Finn, Lou Berger, Craig Gunther, Christophe Mangin and Jouni Korhonen for their various contributions to this work.

10. References

10.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.

- [RFC3031] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, DOI 10.17487/RFC3031, January 2001, <<https://www.rfc-editor.org/info/rfc3031>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

10.2. Informative References

- [I-D.ietf-detnet-architecture]
Finn, N., Thubert, P., Varga, B., and J. Farkas,
"Deterministic Networking Architecture", draft-ietf-detnet-architecture-13 (work in progress), May 2019.
- [I-D.ietf-detnet-data-plane-framework]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane Framework", draft-ietf-detnet-data-plane-framework-02 (work in progress), September 2019.
- [I-D.ietf-detnet-mpls]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: MPLS", draft-ietf-detnet-mpls-01 (work in progress), July 2019.
- [I-D.ietf-detnet-security]
Mizrahi, T., Grossman, E., Hacker, A., Das, S., Dowdell, J., Austad, H., Stanton, K., and N. Finn, "Deterministic Networking (DetNet) Security Considerations", draft-ietf-detnet-security-05 (work in progress), August 2019.
- [I-D.ietf-spring-segment-routing-mpls]
Bashandy, A., Filsfils, C., Previdi, S., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing with MPLS data plane", draft-ietf-spring-segment-routing-mpls-22 (work in progress), May 2019.
- [IEEE802.1AE-2018]
IEEE Standards Association, "IEEE Std 802.1AE-2018 MAC Security (MACsec)", 2018, <<https://ieeexplore.ieee.org/document/8585421>>.

- [IEEE8021CB] Finn, N., "Draft Standard for Local and metropolitan area networks - Seamless Redundancy", IEEE P802.1CB /D2.1 P802.1CB, December 2015, <<http://www.ieee802.org/1/files/private/cb-drafts/d2/802-1CB-d2-1.pdf>>.
- [IEEE8021Q] IEEE 802.1, "Standard for Local and metropolitan area networks--Bridges and Bridged Networks (IEEE Std 802.1Q-2014)", 2014, <<http://standards.ieee.org/about/get/>>.
- [IEEEP8021CBdb] Mangin, C., "Extended Stream identification functions", IEEE P802.1CBdb /D0.2 P802.1CBdb, August 2019, <<http://www.ieee802.org/1/files/private/cb-drafts/d2/802-1CB-d2-1.pdf>>.
- [RFC3985] Bryant, S., Ed. and P. Pate, Ed., "Pseudo Wire Emulation Edge-to-Edge (PWE3) Architecture", RFC 3985, DOI 10.17487/RFC3985, March 2005, <<https://www.rfc-editor.org/info/rfc3985>>.
- [RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <<https://www.rfc-editor.org/info/rfc4301>>.
- [RFC5921] Bocci, M., Ed., Bryant, S., Ed., Frost, D., Ed., Levrau, L., and L. Berger, "A Framework for MPLS in Transport Networks", RFC 5921, DOI 10.17487/RFC5921, July 2010, <<https://www.rfc-editor.org/info/rfc5921>>.

Authors' Addresses

Balazs Varga (editor)
Ericsson
Magyar Tudosok krt. 11.
Budapest 1117
Hungary

Email: balazs.a.varga@ericsson.com

Janos Farkas
Ericsson
Magyar Tudosok krt. 11.
Budapest 1117
Hungary

Email: janos.farkas@ericsson.com

Andrew G. Malis
Independent

Email: agmalis@gmail.com

Stewart Bryant
Futurewei Technologies

Email: stewart.bryant@gmail.com

Don Fedyk
LabN Consulting, L.L.C.

Email: dfedyk@labn.net

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 7, 2020

X. Geng
M. Chen
Huawei Technologies
Y. Ryoo
ETRI
Z. Li
China Mobile
R. Rahman
Cisco Systems
November 04, 2019

Deterministic Networking (DetNet) Configuration YANG Model
draft-ietf-detnet-yang-04

Abstract

This document contains the specification for Deterministic Networking flow configuration YANG Model. The model allows for provisioning of end-to-end DetNet service along the path without dependency on any signaling protocol.

The YANG module defined in this document conforms to the Network Management Datastore Architecture (NMDA).

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].

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 May 7, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Terminologies	4
3. DetNet Service Module	4
3.1. Service Quality	4
3.2. Service Endpoints	4
3.3. Service Encapsulation	5
4. DetNet Configuration Module	5
4.1. DetNet Application Flow Configuration Attributes	5
4.2. DetNet Service Sub-layer Configuration Attributes	5
4.3. DetNet Forwarding Sub-layer Configuration Attributes	6
4.4. DetNet Sub-network Configurations Attributes	6
5. Overview of DetNet YANG	7
5.1. DetNet YANG Considerations	7
5.1.1. DetNet Service YANG Considerations	7
5.1.2. DetNet Configuration YANG Considerations	7
5.2. DetNet YANG Structures	8
5.2.1. DetNet Service YANG Structure	8
5.2.2. DetNet Configuration YANG Structure	8
6. DetNet Service YANG Model	11
7. DetNet Configuration YANG Model	11
8. Open Issues	35
9. IANA Considerations	36
10. Security Considerations	36
11. Acknowledgements	36
12. References	36
12.1. Normative References	36
12.2. Informative References	37
Authors' Addresses	39

1. Introduction

DetNet (Deterministic Networking) provides a capability to carry specified unicast or multicast data flows for real-time applications with extremely low packet loss rates and assured maximum end-to-end delivery latency. A description of the general background and concepts of DetNet can be found in [I-D.ietf-detnet-architecture].

This document defines a YANG model for DetNet based on YANG data types and modeling language defined in [RFC6991] and [RFC7950], which includes DetNet service module and DetNet configuration module, and YANG model for topology discovery is defined in [I-D.ietf-detnet-topology-yang]. DetNet service module is designed for describe characteristics of services being provided for application flows over a network, while the DetNet configuration module is designed for DetNet flow path establishment, flow status reporting, and DetNet functions configuration in order to achieve end-to-end bounded latency and zero congestion loss.

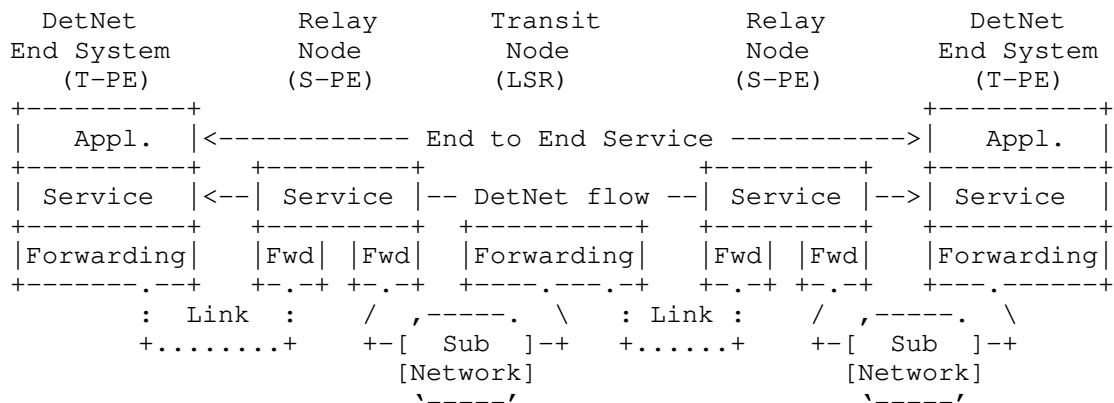


Figure 1: An End-to-end DetNet-Enabled Network

As showed in the picture, in an end-to-end DetNet-enabled network, application flow is carried over a DetNet service and the DetNet service is instantiated as different configuration parameters in different network device along the path. DetNet service is an abstract concept for service provider, and DetNet configuration needs device specific attributes. YANG Models for DetNet service and DetNet configuration are defined in detail respectively in section 3 and section 4.

Editor's notes:

Detnet YANG model and DetNet information model are supposed to keep the same structure and describes the same attributes by different methods. But the design of these two models are still under discussion. The divergence will be settled in the following versions.

2. Terminologies

This documents uses the terminologies defined in [I-D.ietf-detnet-architecture].

3. DetNet Service Module

DetNet Service Module includes service quality attributes, service endpoints attributes and service encapsulation type attributes, which are defined in Section 3.1, 3.2, 3.3 respectively.

3.1. Service Quality

DetNet service quality includes the following attributes:

- o Maximum Latency: MaxLatency is the maximum latency from Ingress to Egress(es) for a single packet of the DetNet flow. MaxLatency is specified as an integer number of nanoseconds.
- o Maximum Latency Variation: MaxLatencyVariation is the difference between the minimum and the maximum end-to-end one-way latency. MaxLatencyVariation is specified as an integer number of nanoseconds.
- o Maximum Loss: MaxLoss defines the maximum Packet Loss Ratio (PLR) parameter for the DetNet service between the Ingress and Egress(es) of the DetNet domain.
- o Maximum Consecutive Loss: Maximum Consecutive Loss defines the consecutive packet loss number.
- o Maximum Misordering: MaxMisordering describes the maximum number of packets that can be received out of order.

3.2. Service Endpoints

Endpoints attribute defines the starting and termination reference points of the DetNet flow by pointing to the ingress interface/node and egress interface(s)/node(s).

3.3. Service Encapsulation

Service Encapsulation attribute defines the data plane type of the DetNet service in a DetNet domain, e.g., MPLS, IP.

Editor's notes: DetNet service module is just defined in the document and the yang model is still under work.

4. DetNet Configuration Module

DetNet configuration module includes DetNet App-flow configuration, DetNet Service Sub-layer configuration, and DetNet Forwarding Sub-layer configuration and DetNet sub-network. The corresponding attributes used in different sub-layers are defined in Section 3.1, 3.2, 3.3, 3.4 respectively.

4.1. DetNet Application Flow Configuration Attributes

DetNet application flow is responsible for mapping between application flows and DetNet flows at the edge node (egress/ingress node). Where the application flows can be either layer 2 or layer 3 flows. To identify a flow at the User Network Interface (UNI), as defined in [I-D.ietf-detnet-flow-information-model], the following flow attributes are introduced:

- o DetNet L3 Flow Identification, refers to Section 7.1.1 of [I-D.ietf-detnet-flow-information-model]
- o DetNet L2 Flow Identification, refers to Section 7.1.2 of [I-D.ietf-detnet-flow-information-model]

Application flow can also do flow filtering and policing at the ingress to prevent the misbehaved flows from going into the network, which needs:

- o Traffic Specification, refers to Section 7.2 of [I-D.ietf-detnet-flow-information-model]

4.2. DetNet Service Sub-layer Configuration Attributes

DetNet service functions, e.g., DetNet tunnel initialization/termination and service protection, are provided in DetNet service sub-layer. To support these functions, the following service attributes need to be configured:

- o DetNet flow identification, refers to Section 8.1.3 of [I-D.ietf-detnet-flow-information-model].

- o Service function indication, indicates which service function will be invoked at a DetNet edge, relay node or end station. (DetNet tunnel initialization or termination are default functions in DetNet service layer, so there is no need for explicit indication.)
- o Flow Rank, refers to Section 8.3 of [I-D.ietf-detnet-flow-information-model].
- o Service Rank, refers to Section 16 of [I-D.ietf-detnet-flow-information-model].
- o Service Sub-layer, refers to Section 4.5 and Section 4.6 of [I-D.ietf-detnet-mpls]
- o Forwarding Sub-layer, refers to Section 4.3 of [I-D.ietf-detnet-ip] and Section 4.5 and Section 4.6 of [I-D.ietf-detnet-mpls]

4.3. DetNet Forwarding Sub-layer Configuration Attributes

As defined in [I-D.ietf-detnet-architecture], DetNet forwarding sub-layer optionally provides congestion protection for DetNet flows over paths provided by the underlying network. Explicit route is another mechanism that is used by DetNet to avoid temporary interruptions caused by the convergence of routing or bridging protocols, and it is also implemented at the DetNet forwarding sub-layer.

To support congestion protection and explicit route, the following transport layer related attributes are necessary:

- o Traffic Specification, refers to Section 7.2 of [I-D.ietf-detnet-flow-information-model]. It may be used for bandwidth reservation, flow shaping, filtering and policing.
- o Explicit path, existing explicit route mechanisms can be reused. For example, if Segment Routing (SR) tunnel is used as the transport tunnel, the configuration is mainly at the ingress node of the transport layer; if the static MPLS tunnel is used as the transport tunnel, the configurations need to be at every transit node along the path; for pure IP based transport tunnel, it's similar to the static MPLS case.

4.4. DetNet Sub-network Configurations Attributes

TBD

5. Overview of DetNet YANG

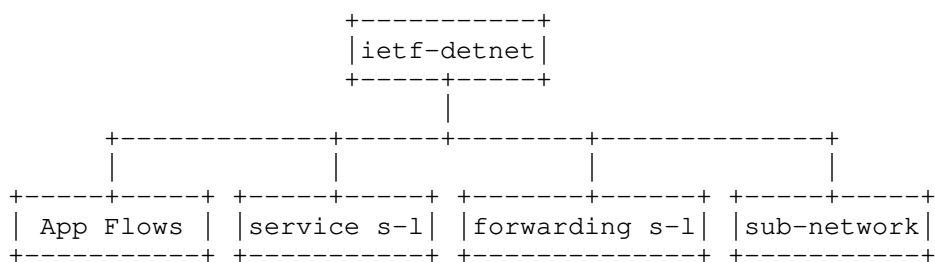
5.1. DetNet YANG Considerations

5.1.1. DetNet Service YANG Considerations

TBD

5.1.2. DetNet Configuration YANG Considerations

The picture shows that the general structure of the DetNet YANG Model:



There are four instances in DetNet YANG Model: App-flow instance, service sub-layer instance, forwarding sub-layer instance and sub-network instance, respectively corresponding to four parts of DetNet functions defined in section 3. In each instance, there are four elements: name, in-segments, out-segments and operations, which means:

- o Name: indicates the key value of the instance identification.
- o In-segments: indicates the key value of identification, e.g., Layer 2 App flow identification, Layer 3 App flow identification and DetNet flow identification.
- o Out-segments: indicates the information of DetNet processing(e.g., DetNet forwarding, DetNet header Encapsulation) and the mapping relationship to the lower sub-layer/sub-network.
- o Operations: indicates DetNet functions, e.g., DetNet forwarding functions, DetNet Service functions, DetNet Resource Reservation.

These elements are different when the technologies used for the specific instance is different. So this structure is abstract, which allows for different technology specifics as defined in different data plane drafts.

5.2. DetNet YANG Structures

5.2.1. DetNet Service YANG Structure

TBD

5.2.2. DetNet Configuration YANG Structure

```

      +--rw app-flow
      |   +--rw operations
      |   |   +--rw sequence-number
      |   |   |   +--rw sequence-number-generation-type?    sequence-number-gener
      |   |   |   +--rw sequence-number-length?              uint8
      |   |   +--rw in-segments
      |   |   |   +--rw app-flow-type?                        flow-type-ref
      |   |   |   +--rw source-mac-address?                  yang:mac-address
      |   |   |   +--rw destination-mac-address?              yang:mac-address
      |   |   |   +--rw ethertype?                            eth:ethertype
      |   |   |   +--rw vlan-id?                              uint16
      |   |   |   +--rw pcp?                                  uint8
      |   |   |   +--rw src-ipv4-prefix                       inet:ipv4-prefix
      |   |   |   +--rw dest-ipv4-prefix                       inet:ipv4-prefix
      |   |   |   +--rw protocol                              uint8
      |   |   |   +--rw dscp?                                  uint8
      |   |   |   +--rw dscp-bitmask?                          uint8
      |   |   |   +--rw src-ipv6-prefix                       inet:ipv6-prefix
      |   |   |   +--rw dest-ipv6-prefix                       inet:ipv6-prefix
      |   |   |   +--rw next-header                           uint8
      |   |   |   +--rw traffic-class?                         uint8
      |   |   |   +--rw traffic-class-bitmask?                uint8
      |   |   |   +--rw flow-label?                            inet:ipv6-flow-label
      |   |   |   +--rw flow-label-flag?                       boolean
      |   |   |   +--rw lower-source-port?                    inet:port-number
      |   |   |   +--rw upper-source-port?                     inet:port-number
      |   |   |   +--rw lower-destination-port?                inet:port-number
      |   |   |   +--rw upper-destination-port?                inet:port-number
      |   |   +--rw out-segments
      |   |   |   +--rw detnet-service-sub-layer?              lower-layer-ref
      |   |   +--rw service-sub-layer
      |   |   |   +--rw operations
      |   |   |   |   +--rw service-operation
      |   |   |   |   |   +--rw service-operation-type?      service-operation-ref
      |   |   |   |   +--rw service-protection
      |   |   |   |   |   +--rw service-protection-type?      service-protection-type
      |   |   |   +--rw in-segments
      |   |   |   |   +--rw detnet-service-type?              flow-type-ref
      |   |   |   |   +--rw detnet-service-list* [detnet-service-index]
      |   |   |   |   |   +--rw detnet-service-index          uint8

```

```

+--rw src-ipv4-prefix                inet:ipv4-prefix
+--rw dest-ipv4-prefix               inet:ipv4-prefix
+--rw protocol                       uint8
+--rw dscp?                          uint8
+--rw dscp-bitmask?                 uint8
+--rw src-ipv6-prefix               inet:ipv6-prefix
+--rw dest-ipv6-prefix              inet:ipv6-prefix
+--rw next-header                    uint8
+--rw traffic-class?                uint8
+--rw traffic-class-bitmask?        uint8
+--rw flow-label?                   inet:ipv6-flow-label
+--rw flow-label-flag?              boolean
+--rw mpls-flow-identification
  +--rw platform-label-flag?         boolean
  +--rw non-platform-label-space
    +--rw incoming-interface?        if:interface-ref
    +--rw non-platform-label-stack* [index]
      +--rw index                    uint8
      +--rw label?                   rt-type:mpls-label
      +--rw tc?                      uint8
  +--rw platform-label-space
    +--rw label?                     rt-type:mpls-label
    +--rw tc?                        uint8
+--rw out-segments
  +--rw detnet-service-processing-type? flow-type-ref
  +--rw detnet-service-encapsulation
    +--rw detnet-service-processing-list* [detnet-service-processi
ng-index]
      +--rw detnet-service-processing-index    uint32
      +--rw ip-flow
        +--rw ipv4-flow
          +--rw src-ipv4-address    inet:ipv4-address
          +--rw dest-ipv4-address    inet:ipv4-address
          +--rw protocol            uint8
          +--rw dscp?               uint8
        +--rw ipv6-flow
          +--rw src-ipv6-address    inet:ipv6-address
          +--rw dest-ipv6-address    inet:ipv6-address
          +--rw next-header          uint8
          +--rw traffic-class?       uint8
          +--rw flow-label?          inet:ipv6-flow-label
        +--rw l4-port-header
          +--rw source-port?         inet:port-number
          +--rw destination-port?    inet:port-number
      +--rw mpls-flow
        +--rw detnet-mpls-label-stack* [index]
          +--rw index                uint8
          +--rw label?               rt-type:mpls-label
          +--rw tc?                  uint8

```

```

      |
      |      +---rw s-bit?                boolean
      |      +---rw d-cw-encapsulate-flag?  boolean
      |      +---rw detnet-forwarding-sub-layer-info
      |      |      +---rw detnet-forwarding-sub-layer?  lower-layer-ref
+---rw forwarding-sub-layer
+---rw operations
+---rw forwarding-operation
|   +---rw forwarding-operation-type?  forwarding-operation-ref
+---rw resource-allocate
|   +---rw interval?                  uint32
|   +---rw max-packets-per-interval?   uint32
|   +---rw max-payload-size?           uint32
|   +---rw average-packets-per-interval? uint32
|   +---rw average-payload-size?       uint32
+---rw qos
+---rw in-segments
+---rw detnet-forwarding-type?         flow-type-ref
+---rw src-ipv4-prefix                 inet:ipv4-prefix
+---rw dest-ipv4-prefix               inet:ipv4-prefix
+---rw protocol                       uint8
+---rw dscp?                          uint8
+---rw dscp-bitmask?                  uint8
+---rw src-ipv6-prefix               inet:ipv6-prefix
+---rw dest-ipv6-prefix              inet:ipv6-prefix
+---rw next-header                    uint8
+---rw traffic-class?                uint8
+---rw traffic-class-bitmask?        uint8
+---rw flow-label?                   inet:ipv6-flow-label
+---rw flow-label-flag?              boolean
+---rw mpls-flow-identification
+---rw platform-label-flag?          boolean
+---rw non-platform-label-space
|   +---rw incoming-interface?        if:interface-ref
|   +---rw non-platform-label-stack* [index]
|   |   +---rw index                  uint8
|   |   +---rw label?                rt-type:mpls-label
|   |   +---rw tc?                   uint8
+---rw platform-label-space
+---rw label?                        rt-type:mpls-label
+---rw tc?                          uint8
+---rw out-segments
+---rw detnet-forwarding-processing-type? flow-type-ref
+---rw natively-detnet-forwarding
|   +---rw ipv4-flow
|   |   +---rw ipv4-next-hop-address?  inet:ipv4-address
|   +---rw ipv6-flow
|   |   +---rw ipv6-next-hop-address?  inet:ipv6-address
+---rw detnet-forwarding-encapsulation

```

```

+--rw ip-flow
|   +--rw ipv4-flow
|   |   +--rw src-ipv4-address      inet:ipv4-address
|   |   +--rw dest-ipv4-address    inet:ipv4-address
|   |   +--rw protocol              uint8
|   |   +--rw dscp?                 uint8
|   +--rw ipv6-flow
|   |   +--rw src-ipv6-address      inet:ipv6-address
|   |   +--rw dest-ipv6-address    inet:ipv6-address
|   |   +--rw next-header           uint8
|   |   +--rw traffic-class?        uint8
|   |   +--rw flow-label?           inet:ipv6-flow-label
|   +--rw l4-port-header
|   |   +--rw source-port?          inet:port-number
|   |   +--rw destination-port?     inet:port-number
+--rw mpls-flow
|   +--rw detnet-mpls-label-stack* [index]
|   |   +--rw index                  uint8
|   |   +--rw label?                 rt-type:mpls-label
|   |   +--rw tc?                    uint8
|   |   +--rw s-bit?                 boolean
|   |   +--rw d-cw-encapsulate-flag? boolean
+--rw lower-layer-info
|   +--rw lower-layer-type?          flow-type-ref
|   +--rw interface
|   |   +--rw outgoing-interface?    if:interface-ref
+--rw sub-layer
|   +--rw sub-layer?                 lower-layer-ref

```

6. DetNet Service YANG Model

TBD

7. DetNet Configuration YANG Model

```

<CODE BEGINS> file ietf-detnet-config@20190324.yang
module ietf-detnet-config {
  namespace "urn:ietf:params:xml:ns:yang:ietf-detnet-config";
  prefix "ietf-detnet";

  import ietf-yang-types {
    prefix "yang";
  }

  import ietf-inet-types{
    prefix "inet";
  }

```

```
import ietf-ethertypes {
  prefix "eth";
}

import ietf-routing-types {
  prefix "rt-type";
}

import ietf-interfaces {
  prefix "if";
}

organization "IETF DetNet Working Group";

contact
  "WG Web:    <http://tools.ietf.org/wg/detnet/>
  WG List:    <mailto:detnet@ietf.org>
  WG Chair:   Lou Berger
              <mailto:lberger@labn.net>

              Janos Farkas
              <mailto:janos.farkas@ericsson.com>

  Editor:     Xuesong Geng
              <mailto:gengxuesong@huawei.com>

  Editor:     Mach Chen
              <mailto:mach.chen@huawei.com>

  Editor:     Zhenqiang Li
              <mailto:lizhenqiang@chinamobile.com>

  Editor:     Reshad Rahman
              <mailto:rrahman@cisco.com>

  Editor:     Yeoncheol Ryoo
              <mailto:dbduscjf@etri.re.kr>";

description
  "This YANG module describes the parameters needed
  for DetNet flow configuration and flow status reporting";

revision 2019-03-24 {
  description "initial revision";
  reference "RFC XXXX: draft-ietf-detnet-yang-02";
}

identity ttl-action {
```

```
    description
      "Base identity from which all TTL
        actions are derived";
  }

  identity no-action {
    base "ttl-action";
    description
      "Do nothing regarding the TTL";
  }

  identity copy-to-inner {
    base "ttl-action";
    description
      "Copy the TTL of the outer header
        to the inner header";
  }

  identity decrease-and-copy-to-inner {
    base "ttl-action";
    description
      "Decrease TTL by one and copy the TTL
        to the inner header";
  }

  identity config-type {
    description
      "Base identity from which all configuration instances are derived";
  }

  identity App-flow {
    base "config-type";
    description
      "App-flow configuration";
  }

  identity service-sub-layer {
    base "config-type";
    description
      "A DetNet MPLS or IP service sub-layer configuration";
  }

  identity forwarding-sub-layer {
    base "config-type";
    description
      "A DetNet MPLS or IP forwarding sub-layer configuration";
  }
```

```
identity tsn-sub-network {
  base "config-type";
  description
    "A TSN sub-net configuration";
}

identity flow-type {
  description
    "Base identity from which all flow type are derived";
}

identity ipv4 {
  base "flow-type";
  description
    "An IPv4 flow";
}

identity ipv6 {
  base "flow-type";
  description
    "An IPv6 flow";
}

identity mpls {
  base "flow-type";
  description
    "An MPLS flow";
}

identity l2 {
  base "flow-type";
  description
    "An MPLS flow";
}

identity tsn {
  base "flow-type";
  description
    "An MPLS flow";
}

identity service-operation {
  description
    "Base identity from which all service operation are derived";
}

identity service-initiation {
  base "service-operation";
}
```



```
    description
      "A DetNet service encapsulates";
  }

  identity service-termination {
    base "service-operation";
    description
      "A DetNet service decapsulates";
  }

  identity service-relay {
    base "service-operation";
    description
      "A DetNet service swap";
  }

  identity forwarding-operation {
    description
      "Base identity from which all data plane operation are derived";
  }

  identity natively-forward {
    base "forwarding-operation";
    description
      "A packet natively forward to lower-layer";
  }

  identity impose-and-forward {
    base "forwarding-operation";
    description
      "Impose a header(MPLS/IP) and forward to lower-layer";
  }

  identity pop-and-forward {
    base "forwarding-operation";
    description
      "Pop an identified packet header and forward to lower-layer";
  }

  identity pop-impose-and-forward {
    base "forwarding-operation";
    description
      "Pop an identified packet header, impose a one or more outgoing
      header and forward to lower-layer ";
  }

  identity swap-and-forward {
    base "forwarding-operation";
```

```
    description
      "Swap an identified packet header with outgoing header and forward
       to lower-layer ";
  }

  identity pop-and-lookup {
    base "forwarding-operation";
    description
      "Pop an identified packet header and perform a lookup";
  }
  identity label-space {
    description
      "Base identity from which all label space are derived";
  }

  identity platform-label {
    base "label-space";
    description
      "label allocated from the platform label space";
  }

  identity non-platform-label {
    base "label-space";
    description
      "label allocated from the non-platform label space";
  }

  typedef ttl-action-definition {
    type identityref {
      base "ttl-action";
    }
    description
      "TTL action definition";
  }

  typedef config-type-ref {
    type identityref {
      base "config-type";
    }
    description
      "config-type-ref";
  }

  typedef flow-type-ref {
    type identityref {
      base "flow-type";
    }
    description
```

```
    "flow-type-ref";
}

typedef service-operation-ref{
    type identityref {
        base "service-operation";
    }
    description
        "service-operation-ref";
}

typedef forwarding-operation-ref {
    type identityref {
        base "forwarding-operation";
    }
    description
        "forwarding-operation-ref";
}

typedef label-space-ref {
    type identityref {
        base "label-space";
    }
    description
        "label-space-ref";
}

typedef lower-layer-ref {
    type leafref {
        path "/ietf-detnet:detnet-config/ietf-detnet:detnet-config-list"
        + "/ietf-detnet:name";
    }
    description
        "lower-layer-ref";
}

typedef service-protection-type {
    type enumeration {
        enum none {
            description
                "no service protection provide";
        }
        enum replication {
            description
                "A Packet Replication Function (PRF) replicates
                DetNet flow packets and forwards them to one or
                more next hops in the DetNet domain. The number
```

```

    of packet copies sent to each next hop is a
    DetNet flow specific parameter at the node doing
    the replication. PRF can be implemented by an
    edge node, a relay node, or an end system";
}
enum elimination {
    description
        "A Packet Elimination Function (PEF) eliminates
        duplicate copies of packets to prevent excess
        packets flooding the network or duplicate
        packets being sent out of the DetNet domain.
        PEF can be implemented by an edge node, a relay
        node, or an end system.";
}
enum ordering {
    description
        "A Packet Ordering Function (POF) re-orders
        packets within a DetNet flow that are received
        out of order. This function can be implemented
        by an edge node, a relay node, or an end system.";
}
enum elimination-ordering {
    description
        "A combination of PEF and POF that can be
        implemented by an edge node, a relay node, or
        an end system.";
}
enum elimination-replication {
    description
        "A combination of PEF and PRF that can be
        implemented by an edge node, a relay node, or
        an end system";
}
enum elimination-ordering-replicaiton {
    description
        "A combination of PEF, POF and PRF that can be
        implemented by an edge node, a relay node, or
        an end system";
}
}
description
    "service-protection-type";
}

typedef sequence-number-generation-type {
    type enumeration {
        enum none {
            description

```

```
        "No sequence number generation function provide";
    }
    enum copy-from-app-flow {
        description
            "Copy the app-flow sequence number to the DetNet-flow";
    }
    enum generate-by-detnet-flow {
        description
            "Generate the sequence number by DetNet flow";
    }
}
description
    "sequence-number-generation-type";
}

grouping l4-port-header {
    description
        "The TCP/UDP port(source/destination) information";
    leaf source-port {
        type inet:port-number;
        description
            "The source port number";
    }
    leaf destination-port {
        type inet:port-number;
        description
            "The destination port number";
    }
}

grouping ipv4-header {
    description
        "The IPv4 packet header information";
    leaf src-ipv4-address {
        type inet:ipv4-address;
        mandatory true;
        description
            "The source IP address of the header";
    }
    leaf dest-ipv4-address {
        type inet:ipv4-address;
        mandatory true;
        description
            "The destination IP address of the header";
    }
    leaf protocol {
        type uint8;
        mandatory true;
    }
}
```

```
        description
            "The protocol of the header";
    }
    leaf dscp {
        type uint8;
        description
            "The DSCP field of the header";
    }
}

grouping ipv6-header {
    description
        "The IPv6 packet header information";
    leaf src-ipv6-address {
        type inet:ipv6-address;
        mandatory true;
        description
            "The source IP address of the header";
    }
    leaf dest-ipv6-address {
        type inet:ipv6-address;
        mandatory true;
        description
            "The destination IP address of the header";
    }
    leaf next-header {
        type uint8;
        mandatory true;
        description
            "The next header of the IPv6 header";
    }
    leaf traffic-class {
        type uint8;
        description
            "The traffic class value of the header";
    }
    leaf flow-label {
        type inet:ipv6-flow-label;
        description
            "The flow label value of the header";
    }
}

grouping mpls-header {
    description
        "The MPLS packet header information";
    leaf label {
        type rt-type:mpls-label;
```

```
        description
            "The label value of the MPLS header";
    }
    leaf tc {
        type uint8;
        description
            "The traffic class value of the MPLS header";
    }
    leaf s-bit {
        type boolean;
        description
            "The s-bit value of the MPLS header,
            which indicates the bottom of the label shack";
    }
    leaf d-cw-encapsulate-flag {
        type boolean;
        description
            "the indication of whether D-CW is encapsulated or not,
            when the D-CW is encapsulated, the sequence number is
            determined by sequence generation type";
    }
}

grouping l2-header {
    description
        "The Ethernet or TSN packet header information";
    leaf source-mac-address {
        type yang:mac-address;
        description
            "The source MAC address value of the ethernet header";
    }
    leaf destination-mac-address {
        type yang:mac-address;
        description
            "The destination MAC address value of the ethernet header";
    }
    leaf ethertype {
        type eth:ethertype;
        description
            "The ethernet packet type value of the ethernet header";
    }
    leaf vlan-id {
        type uint16;
        description
            "The Vlan value of the ethernet header";
    }
    leaf pcp {
        type uint8;
    }
}
```

```
        description
          "The priority value of the ethernet header";
      }
  }

  grouping l4-port-identification {
    description
      "The TCP/UDP port(source/destination) identification information";
    leaf lower-source-port {
      type inet:port-number;
      description
        "The lower source port number of the source port range";
    }
    leaf upper-source-port {
      type inet:port-number;
      description
        "The upper source port number of the source port range";
    }
    leaf lower-destination-port {
      type inet:port-number;
      description
        "The lower destination port number or the destination port range";
    }
    leaf upper-destination-port {
      type inet:port-number;
      description
        "The upper destination port number of the destination port range";
    }
  }

  grouping ipv4-flow-identification {
    description
      "The IPv4 packet header identification information";
    leaf src-ipv4-prefix {
      type inet:ipv4-prefix;
      mandatory true;
      description
        "The source IP address of the header";
    }
    leaf dest-ipv4-prefix {
      type inet:ipv4-prefix;
      mandatory true;
      description
        "The destination IP address of the header";
    }
    leaf protocol {
      type uint8;
      mandatory true;
    }
  }
```



```
        description
            "The protocol of the header";
    }
    leaf dscp {
        type uint8;
        description
            "The DSCP field of the header";
    }
    leaf dscp-bitmask {
        type uint8;
        description
            "The bitmask value that determines whether to use
             the DSCP(IPv4) value for flow identification or not";
    }
}

grouping ipv6-flow-identification {
    description
        "The IPv6 packet header identification information";
    leaf src-ipv6-prefix {
        type inet:ipv6-prefix;
        mandatory true;
        description
            "The source IP address of the header";
    }
    leaf dest-ipv6-prefix {
        type inet:ipv6-prefix;
        mandatory true;
        description
            "The destination IP address of the header";
    }
    leaf next-header {
        type uint8;
        mandatory true;
        description
            "The next header of the IPv6 header";
    }
    leaf traffic-class {
        type uint8;
        description
            "The traffic class value of the header";
    }
    leaf traffic-class-bitmask {
        type uint8;
        description
            "The bitmask value that determines whether to use
             the Traffic class(IPv6) value for flow identification or not";
    }
}
```

```
    leaf flow-label {
      type inet:ipv6-flow-label;
      description
        "The flow label value of the header";
    }
    leaf flow-label-flag {
      type boolean;
      description
        "The flag that determines whether to use
        the Flow Label value for flow identification or not";
    }
  }
}

grouping mpls-flow-identification {
  description
    "The MPLS packet header identification information";
  leaf label {
    type rt-type:mpls-label;
    description
      "The label value of the MPLS header";
  }
  leaf tc {
    type uint8;
    description
      "The traffic class value of the MPLS header";
  }
}

grouping l2-flow-identification {
  description
    "The Ethernet or TSN packet header identification information";
  leaf source-mac-address {
    type yang:mac-address;
    description
      "The source MAC address value of the ethernet header";
  }
  leaf destination-mac-address {
    type yang:mac-address;
    description
      "The destination MAC address value of the ethernet header";
  }
  leaf ethertype {
    type eth:ethertype;
    description
      "The ethernet packet type value of the ethernet header";
  }
  leaf vlan-id {
    type uint16;
  }
}
```

```
        description
            "The Vlan value of the ethernet header";
    }
    leaf pcps {
        type uint8;
        description
            "The priority value of the ethernet header";
    }
}

grouping traffic-specification {
    description
        "traffic-specification specifies how the Source
        transmits packets for the flow. This is the
        promise/request of the Source to the network.
        The network uses this traffic specification
        to allocate resources and adjust queue
        parameters in network nodes.";
    reference
        "draft-ietf-detnet-flow-information-model";
    leaf interval {
        type uint32;
        description
            "The period of time in which the traffic
            specification cannot be exceeded";
    }
    leaf max-packets-per-interval {
        type uint32;
        description
            "The maximum number of packets that the
            source will transmit in one Interval.";
    }
    leaf max-payload-size {
        type uint32;
        description
            "The maximum payload size that the source
            will transmit.";
    }
    leaf average-packets-per-interval {
        type uint32;
        description
            "The average number of packets that the
            source will transmit in one Interval";
    }
    leaf average-payload-size {
        type uint32;
        description
            "The average payload size that the
```

```
        source will transmit.";
    }
}

container detnet-config {
  description
    "DetNet configurations";
  leaf node-id {
    type yang:dotted-quad;
    description
      "A 32-bit number in the form of a dotted quad that is used by
      identifying a DetNet node";
  }
  list detnet-config-list {
    key "name";
    description
      "list of the DetNet configurations";
    leaf name {
      type string;
      description
        "The name to identify the DetNet configuration";
    }
    leaf config-type {
      type config-type-ref;
      description
        "The DetNet configuration type such as a App-flow, service
        sub-layer, forwarding sub-layer, and TSN sub-network";
    }
  }
  container App-flow {
    when "../config-type = 'ietf-detnet:App-flow'";
    description
      "The DetNet App-flow configuration";
    container operations {
      description "operations";
      container sequence-number {
        description "The DetNet sequence number operations grouping";
        leaf sequence-number-generation-type {
          type sequence-number-generation-type;
          description "The DetNet sequence number generation type";
        }
        leaf sequence-number-length {
          type uint8;
          description
            "The DetNet sequence number length";
        }
      }
    }
  }
  container in-segments {
```

```
description "The App-flow identification information";
leaf app-flow-type {
  type flow-type-ref;
  description
    "The App-flow type such as a L2, IPv4, and IPv6";
}
uses l2-flow-identification {
  when "app-flow-type = 'ietf-detnet:tsn' or 'ietf-detnet:l2'";
}
uses ipv4-flow-identification {
  when "app-flow-type = 'ietf-detnet:ipv4'";
}
uses ipv6-flow-identification {
  when "app-flow-type = 'ietf-detnet:ipv6'";
}
uses l4-port-identification {
when "app-flow-type = 'ietf-detnet:ipv6' or 'ietf-detnet:ipv4'";
  or 'ietf-detnet:ipv4'";
}
}
container out-segments {
  description
    "The DetNet service information associated with this App-flow";
  leaf detnet-service-sub-layer {
    type lower-layer-ref;
    description "Specify associated service sub-layer";
  }
}
container service-sub-layer {
  when "../config-type = 'ietf-detnet:service-sub-layer'";
  description "The DetNet service sub-layer configuration";
  container operations {
    description
      "The DetNet service sub-layer operations grouping";
    container service-operation {
      description "The DetNet service operations grouping";
      leaf service-operation-type {
        type service-operation-ref;
        description
          "The DetNet service operations type such as DetNet
            service initiation, termination, and relay";
      }
    }
  }
  container service-protection {
    description
      "The DetNet service protection operations grouping";
    leaf service-protection-type {
```

```

        type service-protection-type;
        description
            "The DetNet service protection type such as PRF, PEF, PEOF,
            PERF, and PEORF";
    }
}
}
container in-segments {
    when "../operations/service-operation"
    + "/service-operation-type != 'service-initiation'";
    description
        "DetNet service identification information";
    leaf detnet-service-type {
        type flow-type-ref;
        description
            "incoming DetNet service flow type";
    }
    list detnet-service-list {
        key "detnet-service-index";
        description
            "Incoming DetNet member flows or a compound flow";
        leaf detnet-service-index {
            type uint8;
            description
                "Incoming DetNet service index";
        }
        uses ipv4-flow-identification {
when "../detnet-service-type = 'ietf-detnet:ipv4'";
        }
        uses ipv6-flow-identification {
when "../detnet-service-type = 'ietf-detnet:ipv6'";
        }
        container mpls-flow-identification {
            when "../detnet-service-type = 'ietf-detnet:mpls'";
            description
                "MPLS type DetNet service identification";
        }
        leaf label-space {
            type label-space-ref;
            description
                "Indicate the incoming MPLS label is associated with
                platform label space or not";
        }
        container non-platform-label-space {
when "../label-space = 'ietf-detnet:non-platform-label'";
            description
                "MPLS label is associated with non-platform label space,
                all of the F-labels and incoming interface information was
                used for identification";
        }
    }
}

```

```
    leaf incoming-interface {
      type if:interface-ref;
      description
        "DetNet service incoming interface information";
    }
    list non-platform-label-stack {
      key "index";
      description
        "All of the label information from the outer label
        to the current label";
      leaf index {
        type uint8;
        description
          "Index of the labels stack";
      }
      uses mpls-flow-identification;
    }
  }
  container platform-label-space {
    when "../label-space = 'ietf-detnet:platform-label'";
    description
      "MPLS label is associated with platform label space, only
      the F-label is used for identification";
    uses mpls-flow-identification;
  }
}

container out-segments {
  when "../operations/service-operation"
  + "/service-operation-type != 'service-termination'";
  description
    "DetNet Service outgoing processing grouping";
  leaf detnet-service-processing-type {
    type flow-type-ref;
    description
      "Outgoing DetNet service flow type";
  }
  container detnet-service-encapsulation {
    description
      "DetNet service encapsulation information";
    list detnet-service-processing-list {
      key "detnet-service-processing-index";
      description
        "The list of single or multiple outgoing DetNet service(s)";
      leaf detnet-service-processing-index {
        type uint32;
        description "Outgoing segment entry";
      }
    }
  }
}
```

```
}
container ip-flow {
  when "../.../detnet-service-processing-type ="
  + "'ietf-detnet:ipv4' or 'ietf-detnet:ipv6'";
  description
    "IP type DetNet flow(s) encapsulation information";
  container ipv4-flow {
    when "../.../detnet-service-processing-type ="
    + "'ietf-detnet:ipv4'";
    description
      "IPv4 packet header encapsulation information";
    uses ipv4-header;
  }
  container ipv6-flow {
    when "../.../detnet-service-processing-type ="
    + "'ietf-detnet:ipv6'";
    description
      "IPv6 packet header encapsulation information";
    uses ipv6-header;
  }
  container l4-port-header {
    description
      "TCP/UDP source or destination port number";
    uses l4-port-header;
  }
}
container mpls-flow {
  when "../.../detnet-service-processing-type ="
  + "'ietf-detnet:mpls'";
  description
    "MPLS type DetNet flow(s) encapsulation information";
  list detnet-mpls-label-stack {
    key "index";
    description
      "The list of MPLS labels stack for swap or encapsulation";
    leaf index {
      type uint8;
      description "Index of the labels stack";
    }
    uses mpls-header;
  }
}
container detnet-forwarding-sub-layer-info {
  description
    "The forwarding sub-layer information that associated with
    this DetNet service sub-layer";
  leaf detnet-forwarding-sub-layer {
    type lower-layer-ref;
  }
}
```



```

        description
            "Specify associated forwarding sub-layer";
    }
}
}
}
}
}
}
container forwarding-sub-layer {
    when "../config-type = 'ietf-detnet:forwarding-sub-layer'";
    description
        "The DetNet forwarding sub-layer configuration";
    container operations {
        description
            "The DetNet forwarding sub-layer operations grouping";
        container forwarding-operation {
            description
                "DetNet forwarding function operations grouping";
            leaf forwarding-operation-type {
                type forwarding-operation-ref;
                description
                    "DetNet forwarding operation type such as
                     natively forward, impose and forward, pop and forward,
                     pop and impose and forward, swap and forward,
                     and pop and lookup";
            }
        }
        container resource-allocate {
            description
                "resource-allocation function operations grouping";
            uses traffic-specification;
        }
        container qos {
            description
                "QoS function operations grouping";
        }
    }
}
container in-segments {
    description
        "DetNet forwarding sub-layer packet identification information";
    leaf detnet-forwarding-type {
        type flow-type-ref;
        description
            "incoming DetNet forwarding packet type";
    }
    uses ipv4-flow-identification {
when "detnet-forwarding-type = 'ietf-detnet:ipv4'";
    }
}

```

```
        uses ipv6-flow-identification {
when "detnet-forwarding-type = 'ietf-detnet:ipv6'";
    }
    container mpls-flow-identification {
        when "../detnet-forwarding-type = 'ietf-detnet:mpls'";
        description
            "MPLS type identification information";
    leaf label-space {
        type label-space-ref;
        description
            "Indicate the incoming MPLS label is associated with platform
            label space or not";
    }
    container non-platform-label-space {
when "../label-space = 'ietf-detnet:non-platform-label'";
        description
            "MPLS label is associated with non-platform label space,
            all of the F-labels and incoming interface information was
            used for identification";
    leaf incoming-interface {
        type if:interface-ref;
        description
            "The information of DetNet forwarding packet incoming
            interface";
    }
    list non-platform-label-stack {
        key "index";
        description
            "All of the label information from the outer label to
            the current label";
    leaf index {
        type uint8;
        description
            "index number 0 indicate last inner label";
    }
        uses mpls-flow-identification;
    }
    }
    container platform-label-space {
        when "../label-space = 'ietf-detnet:platform-label'";
        description
            "MPLS label is associated with platform label space, only
            the F-label is used for identification";
        uses mpls-flow-identification;
    }
    }
    }
    container out-segments {
```

```
description
  "DetNet forwarding sub-layer packet processing information";
leaf detnet-forwarding-processing-type {
  type flow-type-ref;
  description
    "outgoing DetNet forwarding packet type";
}
container natively-detnet-forwarding {
  when "../operations/forwarding-operation"
  + " /forwarding-operation-type = 'natively-forwarding'";
  description
    "Packet forwarding processing information";
  container ipv4-flow {
    when "../detnet-forwarding-processing-type ="
    + "'ietf-detnet:ipv4'";
    description
      "IPv4 type packet forwarding information";
    leaf ipv4-next-hop-address {
      type inet:ipv4-address;
      description
        "IPv4 type Next hop IP address";
    }
  }
  container ipv6-flow {
    when "../detnet-forwarding-processing-type ="
    + "'ietf-detnet:ipv6'";
    description
      "IPv6 type packet forwarding information";
    leaf ipv6-next-hop-address {
      type inet:ipv6-address;
      description
        "IPv6 type Next hop IP address";
    }
  }
}
container detnet-forwarding-encapsulation {
  when "../operations/forwarding-operation"
  + " /forwarding-operation-type != 'natively-forward'";
  description
    "Packet encapsulation information";
  container ip-flow {
    when "../detnet-forwarding-processing-type ="
    + "'ietf-detnet:ipv4' or 'ietf-detnet:ipv6'";
    description
      "The IP type DetNet flow(s) encapsulation information";
    container ipv4-flow {
      when "../detnet-forwarding-processing-type ="
      + "'ietf-detnet:ipv4'";
```

```
        description
            "IPv4 packet header encapsulation information";
        uses ipv4-header;
    }
    container ipv6-flow {
        when "../.../detnet-forwarding-processing-type = "
            + "'ietf-detnet:ipv6'";
        description
            "IPv6 packet header encapsulation information";
        uses ipv6-header;
    }
    container l4-port-header {
        description
            "TCP/UDP source or destination port number";
        uses l4-port-header;
    }
}
container mpls-flow {
    when "../.../detnet-forwarding-processing-type = "
        + "'ietf-detnet:mpls'";
    description
        "MPLS label encapsulation information";
    list detnet-mpls-label-stack {
        key "index";
        description
            "The list of MPLS labels stack for swap or encapsulation";
        leaf index {
            type uint8;
            description
                "Index of the labels stack";
        }
        uses mpls-header;
    }
}
container lower-layer-info {
    description
        "The lower-layer information associated with
        this forwarding sub-layer";
    leaf lower-layer-type {
        type flow-type-ref;
        description
            "indicate lower-layer type";
    }
}
container interface {
    when "../lower-layer-type = 'ietf-detnet:l2'";
    description
        "indicate the lower-layer is the outgoing interface";
    leaf outgoing-interface {
```

```

        type if:interface-ref;
        description
            "Outgoing interface";
    }
}
container sub-layer {
    when "../lower-layer-type != 'ietf-detnet:l2'";
    description
        "indicate the lower-layer is some of the DetNet sub-layer
        or TSN sub-network";
    leaf sub-layer {
        type lower-layer-ref;
        description
            "Specify associated DetNet sub-layer or TSN sub-network";
    }
}
}
}
}
}
container sub-network {
    when "../config-type = 'ietf-detnet:tsn-sub-network'";
    description
        "sub-network";
}
}
}
}

```

<CODE ENDS>

8. Open Issues

There are some open issues that are still under discussion:

- o The Relationship with 802.1 TSN YANG models is TBD. TSN YANG models include: P802.1Qcw, which defines TSN YANG for Qbv, Qbu, and Qci, and P802.1CBcv, which defines YANG for 802.1CB. The possible problem here is how to avoid possible overlap among yang models defined in IETF and IEEE. A common YANG model may be defined in the future to shared by both TSN and DetNet. More discussion are needed here.
- o How to support DetNet OAM is TBD.

These issues will be resolved in the following versions of the draft.

9. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

10. Security Considerations

<TBD>

11. Acknowledgements

12. References

12.1. Normative References

- [I-D.finn-detnet-bounded-latency]
Finn, N., Boudec, J., Mohammadpour, E., Zhang, J., Varga, B., and J. Farkas, "DetNet Bounded Latency", draft-finn-detnet-bounded-latency-04 (work in progress), June 2019.
- [I-D.ietf-detnet-architecture]
Finn, N., Thubert, P., Varga, B., and J. Farkas, "Deterministic Networking Architecture", draft-ietf-detnet-architecture-13 (work in progress), May 2019.
- [I-D.ietf-detnet-flow-information-model]
Farkas, J., Varga, B., Cummings, R., Jiang, Y., and D. Fedyk, "DetNet Flow Information Model", draft-ietf-detnet-flow-information-model-06 (work in progress), October 2019.
- [I-D.ietf-detnet-ip]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: IP", draft-ietf-detnet-ip-03 (work in progress), October 2019.
- [I-D.ietf-detnet-mpls]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: MPLS", draft-ietf-detnet-mpls-03 (work in progress), October 2019.
- [I-D.ietf-detnet-topology-yang]
Geng, X., Chen, M., Li, Z., and R. Rahman, "Deterministic Networking (DetNet) Topology YANG Model", draft-ietf-detnet-topology-yang-00 (work in progress), January 2019.

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 6991, DOI 10.17487/RFC6991, July 2013, <<https://www.rfc-editor.org/info/rfc6991>>.
- [RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <<https://www.rfc-editor.org/info/rfc7950>>.

12.2. Informative References

- [I-D.geng-detnet-info-distribution]
Geng, X., Chen, M., Li, Z., Qin, F., and L. Qiang, "IGP-TE Extensions for DetNet Information Distribution", draft-geng-detnet-info-distribution-04 (work in progress), July 2019.
- [I-D.ietf-detnet-use-cases]
Grossman, E., "Deterministic Networking Use Cases", draft-ietf-detnet-use-cases-20 (work in progress), December 2018.
- [I-D.ietf-teas-yang-te]
Saad, T., Gandhi, R., Liu, X., Beeram, V., and I. Bryskin, "A YANG Data Model for Traffic Engineering Tunnels and Interfaces", draft-ietf-teas-yang-te-21 (work in progress), April 2019.
- [I-D.ietf-teas-yang-te-topo]
Liu, X., Bryskin, I., Beeram, V., Saad, T., Shah, H., and O. Dios, "YANG Data Model for Traffic Engineering (TE) Topologies", draft-ietf-teas-yang-te-topo-22 (work in progress), June 2019.
- [I-D.thubert-tsvwg-detnet-transport]
Thubert, P., "A Transport Layer for Deterministic Networks", draft-thubert-tsvwg-detnet-transport-01 (work in progress), October 2017.
- [I-D.varga-detnet-service-model]
Varga, B. and J. Farkas, "DetNet Service Model", draft-varga-detnet-service-model-02 (work in progress), May 2017.

- [IEEE802.1CB]
IEEE, "IEEE, "Frame Replication and Elimination for Reliability (IEEE Draft P802.1CB)", 2017,
<<http://www.ieee802.org/1/files/private/cb-drafts/>>.", 2016.
- [IEEE802.1Q-2014]
"IEEE, "IEEE Std 802.1Q Bridges and Bridged Networks", 2014, <<http://ieeexplore.ieee.org/document/6991462/>>.", 2014.
- [IEEE802.1Qbu]
IEEE, "IEEE, "IEEE Std 802.1Qbu Bridges and Bridged Networks - Amendment 26: Frame Preemption", 2016,
<<http://ieeexplore.ieee.org/document/7553415/>>.", 2016.
- [IEEE802.1Qbv]
"IEEE, "IEEE Std 802.1Qbu Bridges and Bridged Networks - Amendment 25: Enhancements for Scheduled Traffic", 2015,
<<http://ieeexplore.ieee.org/document/7572858/>>.", 2016.
- [IEEE802.1Qcc]
IEEE, "IEEE, "Stream Reservation Protocol (SRP) Enhancements and Performance Improvements (IEEE Draft P802.1Qcc)", 2017,
<<http://www.ieee802.org/1/files/private/cc-drafts/>>.",
- [IEEE802.1Qch]
IEEE, "IEEE, "Cyclic Queuing and Forwarding (IEEE Draft P802.1Qch)", 2017,
<<http://www.ieee802.org/1/files/private/ch-drafts/>>.", 2016.
- [IEEE802.1Qci]
IEEE, "IEEE, "Per-Stream Filtering and Policing (IEEE Draft P802.1Qci)", 2016,
<<http://www.ieee802.org/1/files/private/ci-drafts/>>.", 2016.
- [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001, <<https://www.rfc-editor.org/info/rfc3209>>.

- [RFC4875] Aggarwal, R., Ed., Papadimitriou, D., Ed., and S. Yasukawa, Ed., "Extensions to Resource Reservation Protocol - Traffic Engineering (RSVP-TE) for Point-to-Multipoint TE Label Switched Paths (LSPs)", RFC 4875, DOI 10.17487/RFC4875, May 2007, <<https://www.rfc-editor.org/info/rfc4875>>.
- [RFC8342] Bjorklund, M., Schoenwaelder, J., Shafer, P., Watsen, K., and R. Wilton, "Network Management Datastore Architecture (NMDA)", RFC 8342, DOI 10.17487/RFC8342, March 2018, <<https://www.rfc-editor.org/info/rfc8342>>.

Authors' Addresses

Xuesong Geng
Huawei Technologies

Email: gengxuesong@huawei.com

Mach(Guoyi) Chen
Huawei Technologies

Email: mach.chen@huawei.com

Yeoncheol Ryoo
ETRI

Email: dbduscjf@etri.re.kr

Zhenqiang Li
China Mobile

Email: lizhenqiang@chinamobile.com

Reshad Rahman
Cisco Systems

Email: rrahman@cisco.com

Network Working Group
Internet-Draft
Intended status: Informational
Expires: January 25, 2020

A. Malis
Independent
X. Geng
M. Chen
Huawei
F. Qin
China Mobile
July 24, 2019

Deterministic Networking (DetNet) Controller Plane Framework
draft-malis-detnet-controller-plane-framework-02

Abstract

This document provides a framework overview for the Deterministic Networking (DetNet) controller plane. It discusses concepts and requirements that will be basis for Detnet controller plane solution documents.

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 January 25, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must

include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
1.1. Terminology	3
2. DetNet Controller Plane Requirements	4
3. DetNet Control Plane Architecture	5
3.1. Distributed Control Plane and Signaling Protocols	5
3.2. SDN/Fully Centralized Control Plane	6
3.3. Hybrid Control Plane	7
4. DetNet Control Plane Additional Details and Issues	8
4.1. Explicit Paths	8
4.2. Resource Reservation	8
4.3. PREOF Support	9
4.4. DetNet in a Traditional MPLS Domain	9
4.5. IP	10
4.6. DetNet with Segment Routing (SR)	10
5. Management Plane Overview	12
5.1. Provisioning	12
5.2. DetNet Operations, Administration and Maintenance (OAM)	12
5.2.1. OAM for Performance Monitoring (PM)	12
5.2.2. OAM for Fault/Defect Management (FM)	13
6. IANA Considerations	13
7. Security Considerations	13
8. Acknowledgments	13
9. References	13
9.1. Normative References	14
9.2. Informative References	15
Authors' Addresses	19

1. Introduction

Deterministic Networking (DetNet) provides the capability to carry specified unicast and/or multicast data flows for real-time applications with extremely low data loss rates and bounded latency within a network domain. As discussed in the Deterministic Networking Architecture [I-D.ietf-detnet-architecture], techniques used to provide this capability include reserving data plane resources for individual (or aggregated) DetNet flows in some or all of the intermediate nodes along the path of the flow, providing explicit routes for DetNet flows that do not immediately change with the network topology, and distributing data from DetNet flow packets over time and/or space to ensure delivery of each packet's data in spite of the loss of a path.

The DetNet data plane is defined in a set of documents that are anchored by the DetNet Data Plane Framework [I-D.ietf-detnet-data-plane-framework] and the associated DetNet MPLS [I-D.ietf-detnet-mpls] and IP [I-D.ietf-detnet-ip] data plane specifications, with additional details and subnet mappings provided in [I-D.ietf-detnet-ip-over-mpls], [I-D.ietf-detnet-mpls-over-udp-ip], [I-D.ietf-detnet-mpls-over-tsn], [I-D.ietf-detnet-ip-over-tsn], and [I-D.ietf-detnet-tsn-vpn-over-mpls].

While the Detnet Architecture and Data Plane Framework documents are primarily concerned with data plane operations, they do contain some references and requirements for functions that would be required in order to automate DetNet service provisioning and monitoring via a DetNet controller plane. The purpose of this document is to gather these references and requirements into a single document and discuss how various possible DetNet controller plane architectures could be used to satisfy these requirements, while not providing the actual protocol details for a DetNet controller plane solution. Such controller plane protocol solutions will be the subject of subsequent documents.

Note that in the DetNet overall architecture, the controller plane includes what are more traditionally considered separate control and management planes. Traditionally, the management plane is primarily involved with node and network provisioning, operational OAM for performance monitoring, and troubleshooting network behaviors and outages, while the control plane is primarily responsible for the instantiation and maintenance of flows, MPLS label allocation and distribution, and active in-band or out-of-band signaling to support these functions. In the DetNet architecture, all of this functionality is combined into a single Controller Plane. See Section 4.4.2 of [I-D.ietf-detnet-architecture] and the aggregation of Control and Management planes in [RFC7426] for further details.

1.1. Terminology

This document uses the terminology established in the DetNet Architecture [I-D.ietf-detnet-architecture], and the reader is assumed to be familiar with that document and its terminology.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. DetNet Controller Plane Requirements

Other DetNet documents, including [I-D.ietf-detnet-architecture] and [I-D.ietf-detnet-data-plane-framework], contain requirements for the Controller Plane. For convenience, these requirements have been compiled here. The primary requirements of the DetNet Controller Plane are that it must be able to:

- o Support the dynamic creation, modification, and deletion of DetNet flows. This may include some or all of explicit path determination, link bandwidth reservations, restricting flows to IEEE 802.1 Time-Sensitive Networking (TSN) links, node buffer and other resource reservations, specification of required queuing disciplines along the path, ability to manage bidirectional flows, etc., as needed for a flow.
- o Support DetNet flow aggregation and de-aggregation via the ability to dynamically create and delete flow aggregates (FAs), and be able to modify existing FAs by adding or deleting members.
- o Operate in a converged network domain that contains both DetNet and non-DetNet flows.
- o Allow flow instantiation requests to originate in an end application (via an Application Programming Interface (API), via static provisioning, or via a dynamic control plane, such as a centralized SDN controller or distributed signaling protocols. See Section 3 for further discussion of these options.
- o In the case of the DetNet MPLS data plane, manage DetNet S-Label and F-Label allocation and distribution.
- o Also in the case of the DetNet MPLS data plane, support packet replication, duplicate elimination, and packet ordering functions (PREOF), and to be able to place these functions at appropriate places in the network.
- o Support applications that require the ability to synchronize the clocks in end systems to the extent supported by the DetNet data plane.
- o Support queue control techniques defined in Section 4.5 of [I-D.ietf-detnet-architecture] and [I-D.finn-detnet-bounded-latency] that require time synchronization among network nodes.
- o Advertise static and dynamic node and link resources such as capabilities and adjacencies to other network nodes (for dynamic

signaling approaches) or to network controllers (for centralized approaches).

- o Adapt to network topology changes such as links or nodes failures.
- o Scale to handle the number of DetNet flows expected in a domain (which may require per-flow signaling or provisioning). This is similar to scalability requirements associated with network slicing [I-D.dong-spring-sr-for-enhanced-vpn].
- o Provision flow identification information at each of the nodes along the path. Flow identification may differ depending on the location in the network and the DetNet functionality (e.g. transit node vs. relay node).
- o Monitor the performance of DetNet flows to ensure that they are meeting required objectives.

3. DetNet Control Plane Architecture

As noted in the Introduction, the DetNet control plane is responsible for the instantiation and maintenance of flows, MPLS label allocation and distribution, and active in-band or out-of-band signaling to support these functions.

The following sections define three possible classes of DetNet control plane architectures: a fully distributed control plane utilizing dynamic signaling protocols, a fully centralized SDN-like control plane, and a hybrid control plane. They discuss the various information exchanges between entities in the network in each of these architectures and the advantages and disadvantages of each option.

In each of the following sections, examples are used to illustrate possible mechanisms that could be used in each of the architectures. These are not meant to be exhaustive or to preclude any other possible mechanism that could be used in place of those used in the examples.

3.1. Distributed Control Plane and Signaling Protocols

In a fully distributed configuration model, User-to-Network Interface (UNI) information is transmitted over a (to-be-defined) DetNet UNI protocol from the user side to the network side, and then UNI and network configuration information propagate in the network via distributed control plane signaling protocols. Using an RSVP-TE traffic-engineered MPLS network as an example:

1. An IGP collects topology information and DetNet capabilities of the network [draft-geng-detnet-info-distribution];
2. The control plane of the ingress edge node receives a flow establishment request from the UNI and calculates one or more valid path(s);
3. Using RSVP-TE [RFC3209], the ingress edge node sends a PATH message with an explicit route. After receiving the PATH message, the egress edge node sends a RESV message with the distributed label and resource reservation request.

Current reservation-oriented distributed control plane protocols, e.g. RSVP-TE and Stream Reservation Protocol (SRP) [IEEE.802.1Qcc-2018], can only reserve bandwidth along the path, while the configuration of a fine-grained schedule, e.g., Time Aware Shaping (TAS) [IEEE.802.1QBV_2015], is not supported. If RSVP-TE or SRP were to be used for a DetNet application, it would require extensions in order to support queue and scheduler reservations in addition to bandwidth reservation.

As discussed in Section 4.9 of [I-D.ietf-detnet-architecture], scalability is a primary concern for DetNet, given the large number of expected flows in a DetNet domain. This could potentially be much larger than, for example, the number of MPLS traffic tunnels in a network using MPLS traffic engineering, which would typically be $N*(N-1)$ tunnels, where N is the number of edge routers in the domain.

Even when flow aggregation is used, DetNet domains can be expected to support a very large number of flows that will need particular queuing disciplines and/or resource allocation, depending on the requirements for each flow. This could require a large amount of dynamic signaling, such as an RSVP-TE session to establish and maintain each flow. Other RSVP-TE scalability concerns are further discussed in [RFC5439].

All of the above tends to argue against a purely distributed control plane for DetNet domains.

3.2. SDN/Fully Centralized Control Plane

In the fully SDN/centralized configuration model, UNI information is transmitted from a Centralized User Configuration (CUC) or from applications via an API or northbound interface to a Centralized Controller, which is the sole source of routing and forwarding information for the domain. Configurations of nodes for DetNet flows are performed by the controller using a protocol such as NETCONF [RFC6241]/YANG [RFC6020] or PCE-CC [RFC8283]. For example:

1. The controller collects topology information and DetNet capabilities of the network via NETCONF/YANG;
2. The controller receives a flow establishment request from a UNI and calculates one or more valid path(s) through the network;
3. The controller chooses the optimal path and configures the devices along that path for flow transmission via PCE-CC.

3.3. Hybrid Control Plane

In the hybrid model, a controller and control plane protocols work together to provide DetNet services, and there are a number of possible combinations. For example:

1. A Centralized Controller collects topology information and DetNet capabilities of the network via an IGP and/or BGP-LS [RFC7752];
2. The controller receives a flow establishment request from a UNI and calculates one or more valid path(s) through the network;
3. Based on the calculation result, the CNC distributes flow path information to the ingress edge node and other information (e.g. replication/duplicate elimination) to the relevant nodes.
4. Using RSVP-TE, the ingress edge node sends a PATH message with an explicit route. After receiving the PATH message, the egress edge node sends a RESV message with the distributed label and resource reservation request.

or

1. The controller collects topology information and DetNet capability of the network via an IGP or BGP-LS;
2. The control plane of the ingress edge node receives a flow establishment request via a UNI;
3. The Ingress edge node sends the path establishment request to the controller through PCEP [RFC5440];
4. After path calculation, the CNC sends the path information of the flow to the ingress edge node via PCEP;
5. Using RSVP-TE, the ingress edge node sends a PATH message with an explicit route. After receiving the PATH message, the egress edge node sends a RESV message with the distributed label and resource reservation request.

There are many other variations that could be included in a hybrid control plane. This document cannot discuss all the possible control plane mechanisms that could be used in hybrid configuration models. Every solution has its own mechanisms and corresponding parameters that are required for it to work.

4. DetNet Control Plane Additional Details and Issues

This section discusses some additional DetNet control plane details and issues.

4.1. Explicit Paths

Explicit paths are required in DetNet to provide a stable transport service and guarantee that DetNet service is not effected when the network topology changes. The following features are necessary to have explicit paths in DetNet:

- o Path computation: DetNet explicit paths need to meet the SLA (Service Level Agreement) requirements and/or resource guarantees from the application/client, which include bandwidth, maximum end-to-end delay, maximum end-to-end delay variation, maximum loss ratio, etc. In an distributed system with IGP-TE, CSPF (Constrained Shortest Path First) can be used to compute a set of feasible paths for a DetNet service. In a system with a network controller, a PCE (Path Computation Engine) can compute paths satisfying the requirements of DetNet with the network information collected from the DetNet domain.
- o Path establishment: Once the path has been computed, the options discussed in Section 3 can be used to establish the path. Also see Section 4.4 and Section 4.6 for some additional considerations depending on the details of the network infrastructure.
- o Strict or loose paths: An explicit path is strict when every intermediate hop is specified so that its route can't change. An explicit path is loose when any IGP route is allowed along the path. Generally, end-to-end SLA guarantees require a strict explicit path in DetNet. However, when the IGP route is known to be able to meet the SLA requirements, loose explicit paths are also acceptable.

4.2. Resource Reservation

Network congestion could cause uncontrolled delay and/or packet loss. DetNet flows are supposed to be protected from congestion, so sufficient resource reservation for DetNet service is necessary. Resources in the network are complex and hard to quantize, and may

include such entities as packet processing resources, packet buffering, port and link bandwidth, and so on. The resources a particular flow requires are determined by the flow's characteristics and SLA.

- o Resource Allocation: Port bandwidth is one of the basic attributes of a network device which is easy to obtain or calculate. In current traffic engineering implementations, network resource allocation is synonymous with bandwidth allocation. A DetNet flow is characterized with a traffic specification as defined in [I-D.ietf-detnet-flow-information-model], including attributes such as Interval, Maximum Packets Per Interval, and Maximum Payload Size. The traffic specification describes the worst case, rather than the average case, for the traffic, to ensure that sufficient bandwidth and buffering resources are reserved to satisfy the traffic specification.
- o Device configuration with or without flow discrimination: The resource allocation can be guaranteed by device configuration. For example, an output port bandwidth reservation can be configured as a parameter of queue management and the port scheduling algorithm. When DetNet flows are aggregated, a group of DetNet flows share the allocated resource in the network device. When the DetNet flows are treated independently, the device should maintain a mapping relationship between a DetNet flow and its corresponding resources.

4.3. PREOF Support

DetNet path redundancy is supported via packet replication and duplicate elimination (PREOF). A DetNet flow is replicated and goes through multiple networks paths to avoid packet loss caused by device or link failures. In general, current control plane mechanisms that can be used to establish an explicit path, whether distributed or centralized, support point-to-point (P2P) and point-to-multipoint (P2MP) path establishment. PREOF requires the ability to compute and establish a point-to-multipoint-to-point (P2MP2P) path. Protocol extensions will be required to support this new feature.

4.4. DetNet in a Traditional MPLS Domain

For the purposes of this document, "traditional MPLS" is defined as MPLS without the use of segment routing (see Section 4.6 for a discussion of MPLS with segment routing) or MPLS-TP [RFC5960].

In traditional MPLS domains, a dynamic control plane using distributed signaling protocols is typically used for the distribution of MPLS labels used for forwarding MPLS packets. The

dynamic signaling protocols most commonly used for label distribution are LDP [RFC5036], RSVP-TE, and BGP [RFC8277] (which enables BGP/MPLS-based Layer 3 VPNs [RFC4384] and Layer 2 VPNs [RFC7432]).

Any of these protocols could be used to distribute DetNet Service Labels (S-Labels) and Aggregation Labels (A-Labels) [I-D.ietf-detnet-mpls]. As discussed in [I-D.ietf-detnet-data-plane-framework], S-Labels are similar to other MPLS service labels, such as pseudowire, L3 VPN, and L2 VPN labels, and could be distributed in a similar manner, such as through the use of targeted LDP or BGP. If these were to be used for DetNet, they would require extensions to support DetNet-specific features such as PREOF, aggregation (A-Labels), node resource allocation, and queue placement.

However, as discussed in Section 3.1, distributed signaling protocols may have difficulty meeting DetNet's scalability requirements. MPLS also allows SDN-like centralized label management and distribution as an alternative to distributed signaling protocols, using protocols such as PCEP and OpenFlow [OPENFLOW].

PCEP, particularly when used as a part of PCE-CC, is a possible candidate protocol to use for centralized management of traditional MPLS-based DetNet domains. However, PCE path calculation algorithms would need to be extended to include the location determination for PREOF nodes in a path, and the means to signal the necessary resource reservation and PREOF function placement information to network nodes. See ((I-D.ietf-pce-pcep-extension-for-pce-controller)) for further discussion of PCE-CC and PCEP for centralized control of an MPLS domain.

4.5. IP

In a later revision of this document, this section will discuss necessary protocol extensions to existing IP routing protocols such as IS-IS and BGP. It should be noted that a DetNet IP domain is simpler than a DetNet MPLS domain, and doesn't support PREOF, so only one path per flow or flow aggregate is required, with no path merging.

4.6. DetNet with Segment Routing (SR)

Segment Routing [RFC8402] is a scalable approach to building network domains that utilizes a combination of source routing in packet headers and centralized network control to compute paths through the network and distribute those paths with associated policy to network edge nodes for use in packet headers. It greatly reduces the amount of network signaling associated with distributed signaling protocols

such as RSVP-TE, and also greatly reduces the amount of state in core nodes compared with that required for traditional MPLS and IP routing, as the state is now in the packets rather than in the routers. This is especially useful for DetNet, where a very large number of flows through a network domain are expected, which would otherwise require the instantiation of state for each flow traversing each node in the network.

The DetNet MPLS and IP data planes were specifically constructed to allow the use of DetNet with both types of segment routing, SR-MPLS [I-D.ietf-spring-segment-routing-mpls] and SRv6 [I-D.ietf-6man-segment-routing-header].

In the DetNet context, DetNet in an SR-MPLS or SRv6 data plane could be used in conjunction with centralized flow management and complete label stack distribution to Detnet domain entry nodes via a centralized controller. Extensions to PCEP to allow the use of PCE-CC with SR-MPLS

One possible architecture is PCE-CC combined with SR-MPLS or SRv6. Extensions to PCEP to allow the use of PCE-CC with SR-MPLS are described in [I-D.zhao-pce-pcep-extension-pce-controller-sr], with SRv6 in [I-D.dhody-pce-pcep-extension-pce-controller-srv6].

This approach would allow the details of packet or flow treatment to be encoded directly in the SIDs on each packet in a flow to reduce the amount of state in network nodes. This approach also allows the integration of DetNet domains with general SR-based backbone networks in a converged domain. In this approach, a new set of functions for DetNet queuing treatments available in the DetNet domain would need to be defined for inclusion in the SR stack.

This is not the only possible approach. There is ongoing work on a number of alternative signaling mechanisms for MPLS-SR and SRv6, including extensions to IGPs and BGP to support distributed signaling. In addition, BGP-LS and BGP route reflectors could be added for a hybrid solution.

A possible mostly centralized hybrid approach could be to use a PCE-CC to push paths represented by SID lists while using BGP-LS to collect network topology and link state information. An IGP is used for the usual link state flooding in order to establish adjacencies, but not for DetNet flow path calculations, only for best effort traffic as usual.

A similar approach for network slicing that could be leveraged for DetNet is described in [I-D.dong-spring-sr-for-enhanced-vpn].

Also, note that SR cannot currently support DetNet PREOF functionality without extensions. One possible approach could be to combine SR with BIER-TE, as discussed in [I-D.ietf-bier-te-arch]. Another possible approach specific to SRv6 is discussed in [I-D.geng-detnet-dp-sol-srv6].

5. Management Plane Overview

The Management Plane includes the ability to statically provision network nodes and to use OAM to monitor DetNet performance and detect outages or other issues at the DetNet layer.

5.1. Provisioning

Static provisioning in a Detnet network will be performed via the use of appropriate YANG models, including [I-D.ietf-detnet-yang] and [I-D.ietf-detnet-topology-yang].

5.2. DetNet Operations, Administration and Maintenance (OAM)

The overall framework and requirements for DetNet OAM are discussed in [I-D.mirsky-detnet-oam]. This document currently includes additional OAM details that may eventually be merged into that document.

5.2.1. OAM for Performance Monitoring (PM)

5.2.1.1. Active PM

Active PM is performed by injecting OAM packets into the network to estimate the performance of the network by measuring the performance of the OAM packets. Adding extra traffic can affect the delay and throughput performance of the network, and for this reason active PM is not recommended for use in operational DetNet domains. However, it is a useful test tool when commissioning a new network.

5.2.1.2. Passive PM

Passive PM monitors the actual service traffic in a network domain in order to measure its performance without having a detrimental affect on the network. As compared to Active PM, Passive PM is much preferred for use in DetNet domains.

A proposal for DetNet passive performance measurement is contained in [I-D.chen-detnet-loss-delay].

5.2.2. OAM for Fault/Defect Management (FM)

[I-D.mirsky-detnet-oam] contains requirements for fault/defect detection and management in a DetNet domain.

6. IANA Considerations

This document has no actions for IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

7. Security Considerations

The overall security considerations of DetNet are discussed in [I-D.ietf-detnet-architecture] and [I-D.ietf-detnet-security]. For DetNet networks that make use of Segment Routing (whether SR-MPLS or SRv6), the security considerations in [RFC8402] also apply.

DetNet networks that make use of a centralized controller plane may be threatened by the loss of connectivity (whether accidental or malicious) between the central controller and the network nodes, and/or the spoofing of control messages from the controller to the network nodes. This is important since such networks depend on centralized controllers to calculate flow paths and instantiate flow state in the network nodes. For networks that use both DetNet and Segment Routing with a centralized controller, this would also include the calculation of SID lists and their installation in edge/border routers.

In both cases, such threats may be mitigated through redundant controllers, the use of authentication between the controller(s) and the network nodes, and other mechanisms for protection against DOS attacks. A mechanism for supporting one or more alternative central controllers and the ability to fail over to such an alternative controller will be required.

8. Acknowledgments

Thanks to Jim Guichard, Donald Eastlake, and Stewart Bryant for their review comments.

9. References

9.1. Normative References

- [I-D.ietf-detnet-architecture]
Finn, N., Thubert, P., Varga, B., and J. Farkas,
"Deterministic Networking Architecture", draft-ietf-
detnet-architecture-13 (work in progress), May 2019.
- [I-D.ietf-detnet-data-plane-framework]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A.,
Bryant, S., and J. Korhonen, "DetNet Data Plane
Framework", draft-ietf-detnet-data-plane-framework-01
(work in progress), July 2019.
- [I-D.ietf-detnet-flow-information-model]
Farkas, J., Varga, B., Cummings, R., and Y. Jiang, "DetNet
Flow Information Model", draft-ietf-detnet-flow-
information-model-04 (work in progress), July 2019.
- [I-D.ietf-detnet-ip]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A.,
Bryant, S., and J. Korhonen, "DetNet Data Plane: IP",
draft-ietf-detnet-ip-01 (work in progress), July 2019.
- [I-D.ietf-detnet-mpls]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A.,
Bryant, S., and J. Korhonen, "DetNet Data Plane: MPLS",
draft-ietf-detnet-mpls-01 (work in progress), July 2019.
- [I-D.ietf-detnet-security]
Mizrahi, T., Grossman, E., Hacker, A., Das, S., Dowdell,
J., Austad, H., Stanton, K., and N. Finn, "Deterministic
Networking (DetNet) Security Considerations", draft-ietf-
detnet-security-04 (work in progress), March 2019.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
Requirement Levels", BCP 14, RFC 2119,
DOI 10.17487/RFC2119, March 1997,
<<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC7426] Haleplidis, E., Ed., Pentikousis, K., Ed., Denazis, S.,
Hadi Salim, J., Meyer, D., and O. Koufopavlou, "Software-
Defined Networking (SDN): Layers and Architecture
Terminology", RFC 7426, DOI 10.17487/RFC7426, January
2015, <<https://www.rfc-editor.org/info/rfc7426>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC
2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174,
May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

- [RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <<https://www.rfc-editor.org/info/rfc8402>>.

9.2. Informative References

- [I-D.chen-detnet-loss-delay]
Chen, M. and A. Malis, "DetNet Packet Loss and Delay Performance Measurement", draft-chen-detnet-loss-delay-01 (work in progress), October 2018.
- [I-D.dhody-pce-pcep-extension-pce-controller-srv6]
Negi, M., Li, Z., and X. Geng, "PCEP Procedures and Protocol Extensions for Using PCE as a Central Controller (PCECC) for SRv6", draft-dhody-pce-pcep-extension-pce-controller-srv6-01 (work in progress), February 2019.
- [I-D.dong-spring-sr-for-enhanced-vpn]
Dong, J., Bryant, S., Miyasaka, T., Zhu, Y., Qin, F., and Z. Li, "Segment Routing for Enhanced VPN Service", draft-dong-spring-sr-for-enhanced-vpn-04 (work in progress), July 2019.
- [I-D.finn-detnet-bounded-latency]
Finn, N., Boudec, J., Mohammadpour, E., Zhang, J., Varga, B., and J. Farkas, "DetNet Bounded Latency", draft-finn-detnet-bounded-latency-04 (work in progress), June 2019.
- [I-D.geng-detnet-dp-sol-srv6]
Geng, X., Chen, M., and Y. Zhu, "DetNet SRv6 Data Plane Encapsulation", draft-geng-detnet-dp-sol-srv6-01 (work in progress), July 2019.
- [I-D.ietf-6man-segment-routing-header]
Filsfils, C., Dukes, D., Previdi, S., Leddy, J., Matsushima, S., and d. daniel.voyer@bell.ca, "IPv6 Segment Routing Header (SRH)", draft-ietf-6man-segment-routing-header-21 (work in progress), June 2019.
- [I-D.ietf-bier-te-arch]
Eckert, T., Cauchie, G., and M. Menth, "Traffic Engineering for Bit Index Explicit Replication (BIER-TE)", draft-ietf-bier-te-arch-03 (work in progress), July 2019.

- [I-D.ietf-detnet-ip-over-mpls]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: IP over MPLS", draft-ietf-detnet-ip-over-mpls-01 (work in progress), July 2019.
- [I-D.ietf-detnet-ip-over-tsn]
Varga, B., Farkas, J., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: IP over IEEE 802.1 Time Sensitive Networking (TSN)", draft-ietf-detnet-ip-over-tsn-00 (work in progress), May 2019.
- [I-D.ietf-detnet-mpls-over-tsn]
Varga, B., Farkas, J., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: MPLS over IEEE 802.1 Time Sensitive Networking (TSN)", draft-ietf-detnet-mpls-over-tsn-00 (work in progress), May 2019.
- [I-D.ietf-detnet-mpls-over-udp-ip]
Varga, B., Farkas, J., Berger, L., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: MPLS over UDP/IP", draft-ietf-detnet-mpls-over-udp-ip-01 (work in progress), July 2019.
- [I-D.ietf-detnet-topology-yang]
Geng, X., Chen, M., Li, Z., and R. Rahman, "Deterministic Networking (DetNet) Topology YANG Model", draft-ietf-detnet-topology-yang-00 (work in progress), January 2019.
- [I-D.ietf-detnet-tsn-vpn-over-mpls]
Varga, B., Farkas, J., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: IEEE 802.1 Time Sensitive Networking over MPLS", draft-ietf-detnet-tsn-vpn-over-mpls-00 (work in progress), May 2019.
- [I-D.ietf-detnet-yang]
Geng, X., Chen, M., Ryoo, Y., Li, Z., and R. Rahman, "Deterministic Networking (DetNet) Configuration YANG Model", draft-ietf-detnet-yang-03 (work in progress), July 2019.
- [I-D.ietf-spring-segment-routing-mpls]
Bashandy, A., Filsfils, C., Previdi, S., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing with MPLS data plane", draft-ietf-spring-segment-routing-mpls-22 (work in progress), May 2019.

- [I-D.mirsky-detnet-oam]
Mirsky, G. and M. Chen, "Operations, Administration and Maintenance (OAM) for Deterministic Networks (DetNet)", draft-mirsky-detnet-oam-03 (work in progress), May 2019.
- [I-D.zhao-pce-pcep-extension-pce-controller-sr]
Zhao, Q., Li, Z., Negi, M., and C. Zhou, "PCEP Procedures and Protocol Extensions for Using PCE as a Central Controller (PCECC) of SR-LSPs", draft-zhao-pce-pcep-extension-pce-controller-sr-05 (work in progress), July 2019.
- [IEEE.802.1QBV_2015]
IEEE, "IEEE Standard for Local and metropolitan area networks -- Bridges and Bridged Networks - Amendment 25: Enhancements for Scheduled Traffic", IEEE 802.1Qbv-2015, DOI 10.1109/IEEESTD.2016.7572858, March 2016, <<http://ieeexplore.ieee.org/servlet/opac?punumber=7572858>>.
- [IEEE.802.1Qcc-2018]
IEEE, "IEEE Standard for Local and Metropolitan Area Networks -- Bridges and Bridged Networks -- Amendment 31: Stream Reservation Protocol (SRP) Enhancements and Performance Improvements", IEEE 802.1Qcc-2018, DOI 10.1109/ieeestd.2018.8514112, October 2018, <<http://ieeexplore.ieee.org/servlet/opac?punumber=8514110>>.
- [OPENFLOW]
Open Networking Foundation, "OpenFlow Switch Specification, Version 1.5.1 (Protocol version 0x06)", ONF TS-025, March 2015, <<https://www.opennetworking.org/wp-content/uploads/2014/10/openflow-switch-v1.5.1.pdf>>.
- [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001, <<https://www.rfc-editor.org/info/rfc3209>>.
- [RFC4384] Meyer, D., "BGP Communities for Data Collection", BCP 114, RFC 4384, DOI 10.17487/RFC4384, February 2006, <<https://www.rfc-editor.org/info/rfc4384>>.
- [RFC5036] Andersson, L., Ed., Minei, I., Ed., and B. Thomas, Ed., "LDP Specification", RFC 5036, DOI 10.17487/RFC5036, October 2007, <<https://www.rfc-editor.org/info/rfc5036>>.

- [RFC5439] Yasukawa, S., Farrel, A., and O. Komolafe, "An Analysis of Scaling Issues in MPLS-TE Core Networks", RFC 5439, DOI 10.17487/RFC5439, February 2009, <<https://www.rfc-editor.org/info/rfc5439>>.
- [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <<https://www.rfc-editor.org/info/rfc5440>>.
- [RFC5960] Frost, D., Ed., Bryant, S., Ed., and M. Bocci, Ed., "MPLS Transport Profile Data Plane Architecture", RFC 5960, DOI 10.17487/RFC5960, August 2010, <<https://www.rfc-editor.org/info/rfc5960>>.
- [RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, DOI 10.17487/RFC6020, October 2010, <<https://www.rfc-editor.org/info/rfc6020>>.
- [RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <<https://www.rfc-editor.org/info/rfc6241>>.
- [RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <<https://www.rfc-editor.org/info/rfc7432>>.
- [RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and S. Ray, "North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP", RFC 7752, DOI 10.17487/RFC7752, March 2016, <<https://www.rfc-editor.org/info/rfc7752>>.
- [RFC8277] Rosen, E., "Using BGP to Bind MPLS Labels to Address Prefixes", RFC 8277, DOI 10.17487/RFC8277, October 2017, <<https://www.rfc-editor.org/info/rfc8277>>.
- [RFC8283] Farrel, A., Ed., Zhao, Q., Ed., Li, Z., and C. Zhou, "An Architecture for Use of PCE and the PCE Communication Protocol (PCEP) in a Network with Central Control", RFC 8283, DOI 10.17487/RFC8283, December 2017, <<https://www.rfc-editor.org/info/rfc8283>>.

Authors' Addresses

Andrew G. Malis
Independent

Email: agmalis@gmail.com

Xuesong Geng
Huawei

Email: gengxuesong@huawei.com

Mach (Guoyi) Chen
Huawei

Email: mach.chen@huawei.com

Fengwei Qin
China Mobile

Email: qinfengwei@chinamobile.com

DetNet
Internet-Draft
Intended status: Standards Track
Expires: May 1, 2020

X. Wang
J. Dai
J. Liu
J. Xu
Fiberhome Telecom LTD
Nov 1, 2019

DetNet Data Plane: IEEE 802.1 Time Sensitive Networking over SRv6
draft-wang-detnet-tsn-over-srv6-00

Abstract

This document specifies the Deterministic Networking data plane when TSN networks interconnected over an Segment Routing IPv6 Packet Switched Networks.

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 May 1, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Terminology	3
2.1. Terms Used in This Document	3
2.2. Abbreviations	3
3. Requirements Language	4
4. IEEE 802.1 TSN Over SRv6 Data Plane Scenario	4
5. IEEE 802.1 TSN Operation Over SRv6 Sub-Networks.	5
5.1. Mapping of TSN Stream ID and Sequence Number	5
5.2. SRv6 Network Programming new Functions	8
5.2.1. End. B.Replication DetNet SID: Packet Replication Function	8
5.2.2. End. B. Elimination: Packet Elimination Function.	9
6. SRv6 Data Plane Considerations	9
6.1. DetNet PREOF	9
6.2. Edge Node Processing	10
7. Management and Control Information Summary.	11
8. Security Considerations	12
9. IANA Considerations	12
10. Acknowledgements	12
11. Normative References.	12
Authors' Addresses	14

1. Introduction

Deterministic Networking (DetNet) is a service that can be offered by a network to DetNet flows. DetNet provides these flows extremely low packet loss rates and assured bounded end-to-end delivery latency. General background and concepts of DetNet can be found in the DetNet Architecture [RFC8655].

Segment Routing(SR) leverages the source routing paradigm. An ingress node steers a packet through an ordered list of instructions, called "segments". SR can be applied over IPv6 data plane using Routing Extension Header(SRH,[I-D.ietf-6man-segment-routing-header]). A segment in Segment Routing is not limited to a routing/forwarding function. A SRv6 Segment can indicate functions that are executed locally in the node where they are defined. [I-D.ietf-spring-srv6-network-programming] describes some well-known functions and segments associated to them. SRH TLVs([I-D.ietf-6man-segment-routing-header]) also provides meta-data for segment processing. All these features make SRv6 suitable to carry DetNet flows, by defining new segments associated with DetNet functions and Meta data for DetNet.

The Time-Sensitive Networking (TSN) is to provide deterministic services through IEEE 802 networks, i.e., guaranteed packet transport with bounded latency, low packet delay variation, and low packet loss. The TSN is a unified industrial Ethernet standard, and supports

production control and information application.

TSN through DetNet needs to focus on the real-time interconnection of multi-subnet network layer. Based on the existing mechanism of TSN, interface scheduling is carried out for routers, firewalls, servers and other devices, in order to ensure the deterministic network services between cross-domain subnets. The remote control requirements across networks of TSN need deterministic transmission of network services through DetNet technology. TSN needs to be deployed with DetNet technology in larger areas such as networking of plant equipment, automatic building control of plant and office buildings.

This document defines how to carry DetNet IEEE 802.1 TSN flows over SRv6 networks.

2. Terminology

2.1. Terms Used in This Document

This document uses the terminology and concepts established in the DetNet architecture [RFC 8655] and [I-D.ietf-detnet-data-plane-framework]. The reader is assumed to be familiar with these documents and their terminology

2.2. Abbreviations

Terminologies for DetNet go along with the definition in [RFC8655]. The following abbreviations are used in this document:

CE: Customer Edge equipment.
CoS: Class of Service.
DetNet: Deterministic Networking.
DF: DetNet Flow.
L2: Layer 2.
L2VPN: Layer 2 Virtual Private Network.
L3: Layer 3.
OAM: Operations, Administration, and Maintenance.
PE: Provider Edge.
PEF: Packet Elimination Function.
PRF: Packet Replication Function.
PREOF: Packet Replication, Elimination and Ordering Functions.
POF: Packet Ordering Function.
QoS: Quality of Service.
TSN: IEEE 802.1 Time-Sensitive Network.
SR: Segment Routing.
SRv6: Segment Routing IPv6.
NH: The IPv6 next-header field.

SID: A Segment Identifier ([RFC8402]).
 SRH: The Segment Routing Header ([I-D.ietf-6man-segment-routing-header]).

3. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

4. IEEE 802.1 TSN over SRv6 Data Plane Scenario

Figure 1 illustrates how DetNet can provide services for IEEE 802.1 TSN end systems, CE1 and CE2, over a DetNet enabled SRv6 network. DetNet Edge Nodes sit at the boundary of a DetNet domain. They are responsible for mapping non-DetNet aware L2 traffic to DetNet services. They also support the imposition and disposition of the required DetNet encapsulation. They understand and support IEEE 802.1 TSN and are able to map TSN flows into DetNet flows. Edge nodes, PE1 and PE2, insert and remove required DetNet SRv6 data plane encapsulation. The 'X' in the edge nodes and relay node, R1, represent a potential DetNet compound flow packet replication and elimination point.

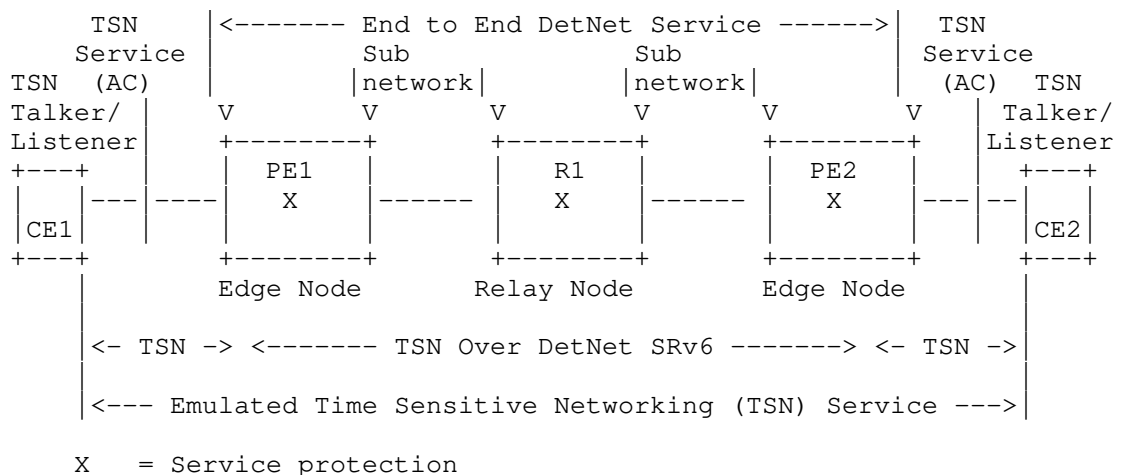


Figure 1: IEEE 802.1 TSN Over DetNet SRv6

Native TSN flow and DetNet SRv6 flow differ not only by the additional SRH specific encapsulation, but DetNet SRv6 flows have on each DetNet node an associated DetNet specific data structure, what

defines flow related characteristics and required forwarding functions. In this example, edge Nodes provide a service proxy function that "associates" the DetNet flows and native flows at the edge of the DetNet domain. This ensures that the DetNet SRv6 Flow is properly served at the Edge node (and inside the domain).

5. IEEE 802.1 TSN Operation Over SRv6 Sub-Networks

A classical SRv6 data plane solution is showed in the picture below:

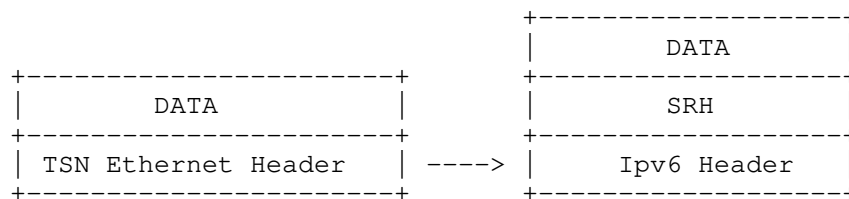


Figure 2: SRv6 DetNet data plane solution

In SRv6 for DetNet, the DATA with the SRH is used for carrying DetNet flows. Traffic Engineering is instantiated in the segment list of SRH, and other functions and arguments for service protection (packet replication, elimination and ordering) and congestion control (packet queuing and forwarding) are also defined in the SRH.

The Time-Sensitive Networking (TSN) Task Group of the IEEE 802.1 Working Group have defined (and are defining) a number of amendments to IEEE 802.1Q [IEEE8021Q] that provides zero congestion loss and bounded latency in bridged networks. Furthermore IEEE 802.1CB [IEEE8021CB] defines frame replication and elimination functions for reliability that should prove both compatible with and useful to, DetNet networks. All these functions have to identify flows those require TSN treatment.

The challenge for SRv6 flows is that the protocol interworking function defined in IEEE 802.1CB [IEEE8021CB] does not work for segment list of SRH flows. The aim of the protocol interworking function is to convert a TSN ingress flow (for examples, identified by a specific destination MAC address and VLAN) to segment list of SRH. A similar interworking pair at the other end of the SRv6 sub-network would restore the packet to its original TSN packet.

The TSN layer 2 header and application payload carried by the TSN network are encapsulated in 'DATA' field of figure 2.

5.1. Mapping of TSN Stream ID and Sequence Number

The Edge node MUST provide the SRv6 sub-network specific segment list of SRH encapsulation over the link(s) towards the sub-network. A SRv6-aware edge node MUST support the following TSN components:

1. For recognizing flows:
 - * Stream Identification (SRv6-flow-aware)
2. For FRER used inside the TSN domain, additionally:
 - * Sequencing function (SRv6-flow-aware)
 - * Sequence encode/decode function
3. For FRER when the node is a TSN replication or elimination point, additionally:
 - * Stream splitting function
 - * Individual recovery function

The Time-Sensitive Networking (TSN) Task Group of the IEEE 802.1 Working Group has defined Stream identification in section 6.1 of IEEE 802.1CB [IEEE8021CB]. Four specific Stream identification functions are described: Null Stream identification, Source MAC and VLAN Stream identification, Active Destination MAC and VLAN Stream identification, and IP Stream identification. These Stream identification functions are summarized as follow:

- o Null Stream identification: destination MAC address, vlan identifier.
- o Source MAC and VLAN Stream identification: source MAC address, vlan identifier.
- o Active Destination MAC and VLAN Stream identification: destination MAC address, vlan identifier.
- o IP Stream identification: destination MAC address, vlan identifier, IP source address, IP destination address, DSCP, IP next protocol, source port, destination port.

The SRH for DetNet in the IPv6 header is showed as follows, according to [I-D.ietf-6man-segment-routing-header] and [I-D.ietf-spring-srv6-network-programming]:

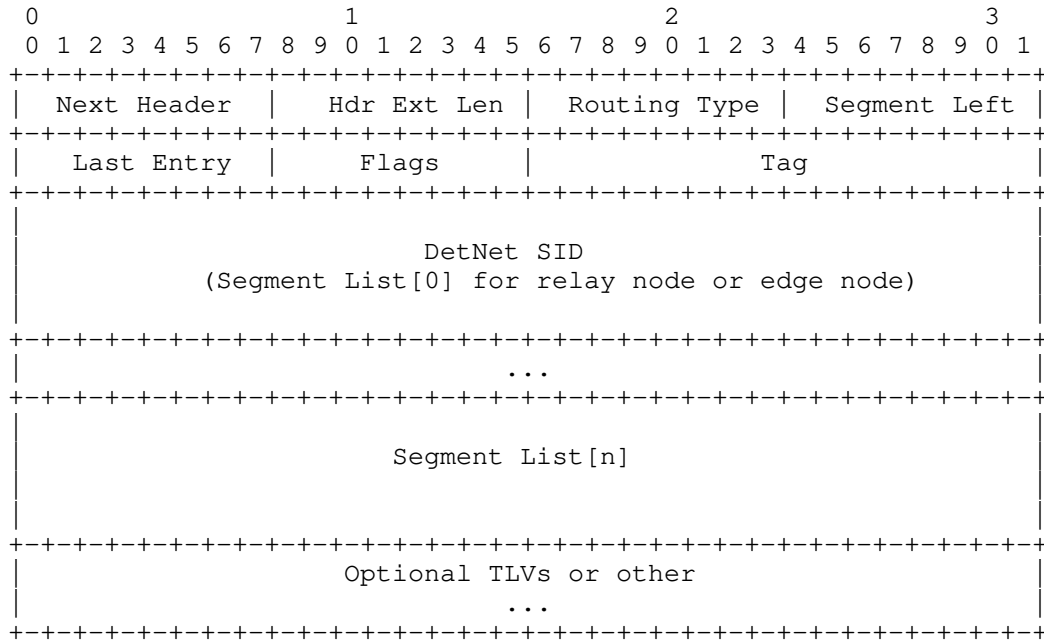


Figure 3: SRH for DetNet

The DetNet SRv6 flow is identified by DetNet SID in SRH. DetNet SID is defined as a 128-bit value.

A new DetNet SID is defined to support DetNet service protection for TSN stream. It is used to uniquely identify a DetNet flow in a SRv6 DetNet node and to discriminate packets in the same DetNet flow by sequence number. DetNet SID is defined as follows:

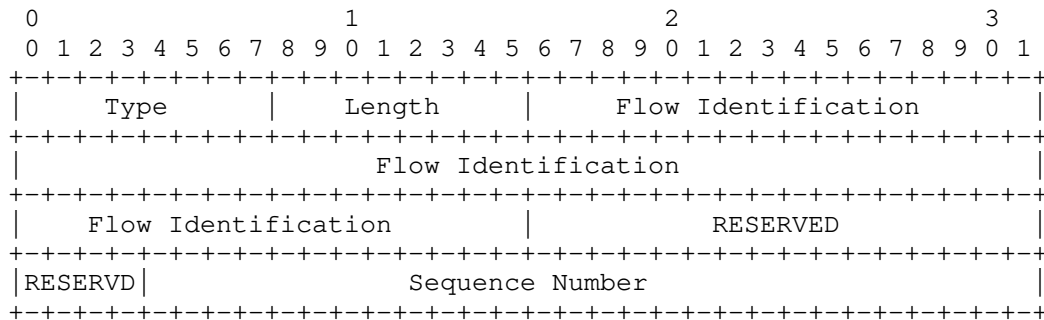


Figure 4: DetNet SID for Flow Identification

Where:

- o Type: 8bits, to be assigned by IANA.
- o Length: 8 bits.
- o Flow Identification: 64 bits, which is used for identifying DetNet flow.
- o RESERVED: 20 bits, MUST be 0 on transmission and ignored on receipt.
- o Sequence Number: 28 bits, which is used for indicating sequence number of a DetNet flow.

When TSN stream is transmitted over a SRv6 network, TSN Stream Identification MUST pair SRv6 flows and TSN Streams and encode that in data plane formats as well. When the new DetNet SID is used to identify DetNet flow and the mapping for TSN stream is as follows:

- o Type: 8bits, to be assigned by IANA, used to identify sources from the TSN stream.
- o Length: 8 bits, the value is 16 octets.
- o Flow Identification: 64 bits, which is used for identifying DetNet flow. The former 48 bit corresponds to the MAC address identified by the TSN stream, and the post 16 bit comes from the VLAN-ID and priority parameters in TSN packet.
- o RESERVED: 20 bits, MUST be 0 on transmission and ignored on receipt.
- o Sequence Number: 28 bits, which is used for indicating sequence number of a DetNet flow. The value comes from the Redundancy tag (R-TAG) in TSN packet as defined in Clause 7.8 of IEEE 802.1CB [IEEE8021CB].

Flow Identification in SRH can identify Null Stream, Source MAC and VLAN Stream, Active Destination MAC and VLAN Stream in TSN stream. For TSN IP Stream, destination MAC address and vlan is still indicated by flow Identification, other IP-based fields correspond to IP fields in SRv6 one by one, such as IP source address, IP destination address, DSCP, IP next protocol, source port, destination port etc.

5.2. SRv6 Network Programming new Functions

New SRv6 Network Programming functions are defined as follows:

5.2.1. End. B.Replication DetNet SID: Packet Replication Function

When N receives a packet whose IPv6 DA is S and S is a local End.B. Replication DetNet SID, does:

S01. IF NH=SRH & SL>0 THEN {

- S02. Extract the DetNet SID values from the SRH or TSN Stream identification and TSN Rtag.
- S03. Create two new outer IPv6+SRH headers: IPv6-SRH-1 and IPv6-SRH-2 Insert the policy-instructed segment lists in each newly created SRH (SRH-1 and SRH-2). Also, add the extracted DetNet SID into SRH-1 and SRH-2.
- S04. Remove the incoming outer IPv6+SRH header, restore DATA as the original packet.
- S05. Create a duplication of the restore DATA as the duplicate packet.
- S06. Encapsulate the original packet into the first outer IPv6+SRH header: (IPv6-SRH-1) (original packet)
- S07. Encapsulate the duplicate packet into the second outer IPv6+SRH header: (IPv6-SRH-2) (duplicate packet)
- S08. Set the IPv6 SA as the local address of this node.
- S09. Set the IPv6 DA of IPv6-SRH-1 to the first segment of the SRv6 Policy in of SRH-1 segment list.
- S10. Set the IPv6 DA of IPv6-SRH-2 to the first segment of the SRv6 Policy in of SRH-2 segment list.
- S11. }

5.2.2. End. B. Elimination: Packet Elimination Function

When N receives a packet whose IPv6 DA is S and S is a local End.B. Elimination DetNet SID, does:

- S01. IF NH=SRH & SL>0 & "the packet is not a redundant packet" THEN {
- S02. Do not decrement SL nor update the IPv6 DA with SRH[SL]
- S03. Extract the value of DetNet SID from the SRH
- S04. Extract Flow Identification and Sequence Number from DetNet SID.
- S05. IF NOT receive the packet with the same Flow Identification and Sequence Number {
- S06. Create a new outer IPv6+SRH header
- S07. Insert the policy-instructed segment lists in the newly created SRH and add the retrieved DetNet SID in the newly created SRH
- S08. Remove the incoming outer IPv6+SRH header.
- S09. Set the IPv6 DA to the first segment of the SRv6 Policy in the newly created SRH
- S10. } Else {
- S11. Drop the packet
- S12. }
- S13. }

6. SRv6 Data Plane Considerations

6.1. DetNet PREOF

Flow Identification and sequence number are necessary in the encapsulation of SRv6 for DetNet in order to support service protection. Replication nodes decide which DetNet flows are supposed to be replicated by the flow identification. Elimination nodes decide whether a packet should be dropped because of redundancy by the flow identification and sequence number.

FRER function and the provided service recovery is available in that the Stream-ID and the TSN sequence number are paired with the SRv6 flow parameters they can be combined with PREOF functions.

SRv6 supporting DetNet flows may use Packet Replication, Elimination and Ordering Functions (PREOF) based on the DetNet SID in SRH, which is derived from TSN Stream. The specific operation of Frame Replication and Elimination for Redundancy (FRER) [802.1CB] is not modified by the use of DetNet and follows IEEE 802.1CB [IEEE8021CB].

6.2. Edge Node Processing

An edge node is responsible for matching ingress packets to the service they require and encapsulating them accordingly. An edge node is a SRv6 DetNet-aware forwarder, and may participate in the packet replication and duplication elimination.

The Controller sends Detnet SRv6 policies to the edge node. These policies include mapping of ingress TSN stream to DetNet SRv6 flow. The detnet SID is associated with an SR Policy, and its value comes from a TSN packet. When the edge node forwards a TSN packet to SRv6 network, inserting an SRH with the policy and adds an outer IPv6 header. The TSN flow identification and sequence number is copied to DetNet SID in SRv6 SRH.

Additionally the DetNet-aware edge node does duplicate frame elimination based on the flow identification and the sequence number combination. The packet replication is also done within the DetNet-aware forwarder. During elimination and the replication process the sequence number of the DetNet member flow MUST be preserved and copied to the egress DetNet member flow.

7. Management and Control Information Summary

The following summarizes the set of information that is needed to support TSN over SRv6 at the ingress edge node:

- o TSN Stream identification and TSN R-tag information to be mapped to SRv6 SRH SID. Note that a single TSN Stream identification can map to one SRH DetNet SID, and it can be used for PREOF.
- o IPv6 source address.

- o IPv6 destination address.
- o IPv6 Traffic Class.

This information MUST be provisioned per DetNet flow via configuration, e.g., via the controller or management plane.

It is the responsibility of the DetNet controller plane to properly provision both flow identification information and the flow specific resources needed to be provided the traffic treatment needed to meet each flow's service requirements. This applies for aggregated and individual flows.

DetNet SRv6 flow and TSN Stream mapping related information are required only for DetNet SRv6 edge nodes; the edge node is TSN-aware and DetNet SRv6-aware node. These DetNet SRv6 edge nodes are member of both the DetNet SRv6 domain and the TSN sub-network. Within the TSN sub-network the DetNet SRv6 node may have a TSM-aware role, so TSN specific management and control plane functionalities must be implemented. There are many similarities in the management plane techniques used in DetNet and TSN, but that is not the case for the control plane protocols. For example, RSVP-TE and MSRP behaves differently. Therefore management and control plane design is an important aspect of scenarios, where mapping between DetNet SRv6 and TSN is required.

In order to use a DetNet SRv6 sub-network between TSN nodes, TSN stream specific information must be converted to SRv6 DetNet SRH. TSN Stream ID and stream related parameters/requirements must be converted to a SRv6 DetNet flow ID and flow related parameters/requirements. Note that, as the DetNet SRv6 sub-network is just a portion of the end2end TSN path (i.e., single hop from IP perspective), some parameters (e.g., delay) may differ significantly. Other TSN stream parameters (like bandwidth) also may have to be tuned due to the SRv6 encapsulation used in the DetNet sub-network.

In some case it may be challenging to determine some TSN Stream related information. For example which DetNet SRv6 paths are multi-Listener of the mapped TSN stream to one TSN stream Talker? However it may be not trivial to locate the point/interface where that Listener is connected to the TSN sub-network. Such attributes may require interaction between control and management plane functions and between DetNet SRv6 and TSN domains.

Mapping between DetNet SRv6 flow identifiers and TSN Stream identifiers, if not provided explicitly, can be done by a DetNet SRv6 node locally based on the configuration of SRv6 Behaviors associated with a SID.

8. Security Considerations

The security considerations of DetNet in general are discussed in [RFC8655] and [I-D.sdt-detnet-security]. Other security considerations will be added in a future version of this draft.

9. IANA Considerations

This document requests assigning new DetNet SID TLV code-points as described in section 5.

10. Acknowledgements

Thanks for Guanghua Lan and Ximing Dong for their comments and contributions.

11. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <<https://www.rfc-editor.org/info/rfc8402>>.
- [I-D.ietf-spring-srv6-network-programming] Filsfils, C., Camarillo, P., Leddy, J., daniel.voyer@bell.ca, d., Matsushima, S., and Z. Li, "SRv6 Network Programming", draft-filsfils-spring-srv6-network-programming-07 (work in progress), February 2019.
- [I-D.ietf-6man-segment-routing-header] Filsfils, C., Dukes, D., Previdi, S., Leddy, J., Matsushima, S., and d. daniel.voyer@bell.ca, "IPv6 Segment Routing Header (SRH)", draft-ietf-6man-segment-routing-header-21 (work in progress), June 2019.
- [RFC8655] Finn, N., Thubert, P., Varga, B., and J. Farkas, "Deterministic Networking Architecture", RFC 8655, May 2019, <<https://www.rfc-editor.org/info/rfc8655>>.
- [I-D.ietf-detnet-dp-sol-mpls] Korhonen, J. and B. Varga, "DetNet MPLS Data Plane Encapsulation", draft-ietf-detnet-dp-sol-mpls-02 (work in progress), June 2019.

progress), March 2019.

[I-D.ietf-spring-segment-routing-policy]

Filsfils, C., Sivabalan, S., daniel.voyer@bell.ca, d., bogdanov@google.com, b., and P. Mattes, "Segment Routing Policy Architecture", draft-ietf-spring-segment-routing-policy-03 (work in progress), May 2019.

[I-D.ietf-detnet-flow-information-model]

Farkas, J., Varga, B., Cummings, R., and Y. Jiang, "DetNet Flow Information Model", draft-ietf-detnet-flow-information-model-03 (work in progress), March 2019.

[I-D.ietf-spring-segment-routing-mpls]

Bashandy, A., Filsfils, C., Previdi, S., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing with MPLS data plane", draft-ietf-spring-segment-routing-mpls-22 (work in progress), May 2019.

[I-D.sdt-detnet-security]

Mizrahi, T., Grossman, E., Hacker, A., Das, S., "Deterministic Networking (DetNet) Security Considerations, draft-sdt-detnet-security, work in progress", 2017.

[I-D.ietf-detnet-ip-over-mpls]

Varga, B., Farkas, J., Berger, L., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: IP over MPLS", draft-ietf-detnet-ip-over-mpls-01 (work in progress), July 2019.

[I-D.ietf-detnet-ip-over-tsn]

Varga, B., Farkas, J., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: IP over IEEE 802.1 Time Sensitive Networking (TSN)", draft-ietf-detnet-ip-over-tsn-00 (work in progress), May 2019.

[I-D.ietf-detnet-mpls-over-tsn]

Farkas, J., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: MPLS over IEEE 802.1 Time Sensitive Networking (TSN)", draft-ietf-detnet-mpls-over-tsn-00 (work in progress), May 2019.

[I-D.ietf-detnet-mpls-over-udp-ip]

Varga, B., Farkas, J., Beger, L., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: MPLS over UDP/IP", draft-ietf-detnet-mpls-over-udp-ip-01 (work in progress), May 2019.

[I-D.ietf-detnet-tsn-vpn-over-mpls]
Varga, B., Farkas, J., Malis, A., Bryant, S., and J.
Korhonen, "DetNet Data Plane: IEEE 802.1 Time Sensitive
Networking over MPLS", draft-ietf-detnet-tsn-vpn-overmpls-
00 (work in progress), May 2019.

[I-D.ietf-geng-detnet-dp-sol-srv6]
Geng, X., Chen, M., Zhu, Y.
"DetNet SRv6 Data Plane Encapsulation", draft-geng-detnet-
dp-sol-srv6-01 (work in progress), May 2019.

[I-D.ietf-geng-spring-srv6-for-detnet]
Geng, X., Li, Z., Chen, M.
"SRv6 for Deterministic Networking (DetNet)", draft-geng-
spring-srv6-for-detnet-00 (work in progress), July 2019.

[IEEE8021CB]
Finn, N., "Draft Standard for Local and metropolitan area
networks - Seamless Redundancy", IEEE P802.1CB
/D2.1 P802.1CB, December 2015,
<[http://www.ieee802.org/1/files/private/cb-drafts/
d2/802-1CB-d2-1.pdf](http://www.ieee802.org/1/files/private/cb-drafts/d2/802-1CB-d2-1.pdf)>.

[IEEE8021Q]
IEEE 802.1, "Standard for Local and metropolitan area
networks--Bridges and Bridged Networks (IEEE Std 802.1Q-
2014)", 2014, <<http://standards.ieee.org/about/get/>>.

Authors' Addresses

Xueshun Wang
Fiberhome Telecom LTD
Email: xswang@fiberhome.com

Jinyou Dai
Fiberhome Telecom LTD
Email: djy@fiberhome.com

Jianhua Liu
Fiberhome Telecom LTD
Email: liujianhua@fiberhome.com

Jing Xu
Fiberhome Telecom LTD
Email: xujing2010@fiberhome.com

DeNet WG
Internet-Draft
Intended status: Standards Track
Expires: May 4, 2020

Q. Xiong
J. Yu
ZTE Corporation
P. Liu
F. Qin
China Mobile
November 1, 2019

DetNet QoS Policy
draft-xiong-detnet-qos-policy-02

Abstract

This document proposes a Quality of Service (QoS) policy to apply Differentiated Services (DiffServ) model for Deterministic Networking (DetNet) and defines a DetNet DiffServ mechanism including DetNet IP and MPLS encapsulation.

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 May 4, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
1.1. Requirements Language	3
1.2. Terminology	3
2. DetNet DiffServ Overview	3
2.1. DetNet Classifiers	4
2.2. DetNet Traffic Conditioners	4
2.2.1. Scheduler	5
2.2.2. Order	5
2.3. DetNet DSCP	5
2.4. DetNet PHB	5
2.5. DetNet Queuing	6
3. DetNet IP DiffServ Consideration	6
4. DetNet MPLS DiffServ Consideration	6
5. Security Considerations	7
6. IANA Considerations	7
7. Acknowledgements	7
8. References	7
8.1. Informative References	7
8.2. Normative References	7
Authors' Addresses	8

1. Introduction

As defined in [RFC8655], Deterministic Networking (DetNet) provides a capability to carry specified unicast or multicast data flows for real-time applications with extremely low data loss rates and bounded latency. DetNet and non-DetNet packets may be allowed to be transmitted in the same network and more than one DetNet flows which have different priorities may be forwarded through the DetNet domain. The DetNet Class of Service (CoS) should be taken into consideration to provide Quality of Service (QoS) for DetNet services.

As discussed in [I-D.ietf-detnet-mpls] and [I-D.ietf-detnet-ip], Differentiated Services (DiffServ) can provide traffic forwarding treatment for DetNet networks. The DiffServ architecture as specified in [RFC2475] defined a model that traffic entering a DiffServ domain is classified and conditioned at the boundaries and marked with a DiffServ Code Point (DSCP) defined in [RFC2474]. The DSCP is used at transit nodes to select the Per Hop Behavior (PHB) that determines the scheduling treatment. And [RFC3270] provides a solution to support DiffServ for traffic marked with Traffic Class (TC) [RFC5462] transported over an MPLS network.

This document proposes a QoS policy to apply DiffServ model for DetNet networks and defines a DetNet DiffServ mechanism including DetNet IP and MPLS encapsulation.

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 [RFC2119].

1.2. Terminology

The terminology is defined as [RFC8655], [RFC3270], [RFC2475] and [RFC2474].

2. DetNet DiffServ Overview

The DetNet network needs to be capable of supporting differentiated services dividing to one or more contiguous DiffServ domains. The key components within a DiffServ domain including traffic classification and conditioning functions, and PHB-based forwarding. The customers may specify packet classification policy, traffic profiles and actions to DetNet flows which are in-profile or out-of-profile at the boundary. The DiffServ domains may support different PHB groups internally and different codepoint->PHB mappings at the transit nodes. The DetNet DiffServ process for packets is as Figure 1 shown.

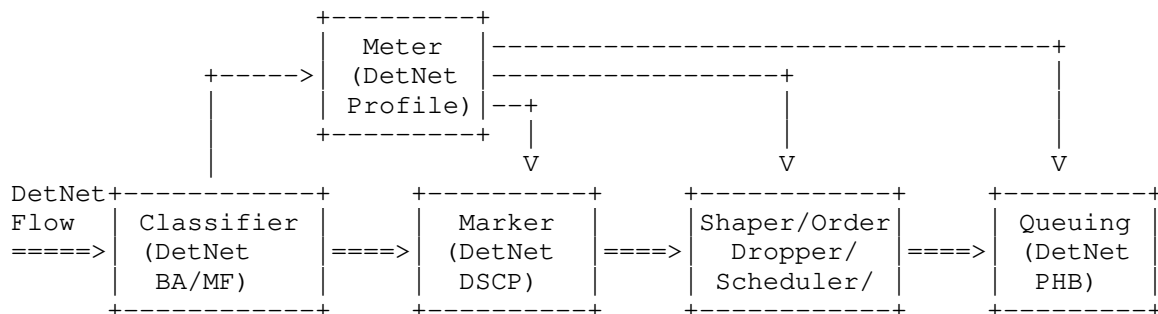


Figure 1: Overview of a DetNet DiffServ mechanism

2.1. DetNet Classifiers

As defined in [RFC2475], packet classifiers select packets in a traffic stream based on the information of packet header including two types of classifiers, the BA (Behavior Aggregate) and MF (Multi-Field) Classifier. The difference is that the BA classifies packets based on the CoS field and the latter one based on more other header fields.

In DetNet DiffServ model, BA and MF can be applied for packets classification. After classification, the flows can be separated from DetNet and non-DetNet. As specified in [I-D.ietf-detnet-ip], no DetNet specific encapsulation is defined to support DetNet IP flow identification and DetNet service delivery. So the DetNet IP classifiers is the same as defined in [RFC2474] and [RFC2475]. As defined in [I-D.ietf-detnet-mpls], DetNet service Label (S-Label) is used to identify a DetNet flow and forwarding labels (F-Labels) are used to provide LSP-based connectivity in DetNet MPLS header. The S-Label and F-Labels can be used in combination with MPLS TC filed in MF classifier. And DetNet MPLS BA classifier select packets based on the MPLS TC field only as defined in [RFC5462].

2.2. DetNet Traffic Conditioners

As mentioned in [RFC8655], DetNet flows can be shaped or scheduled. The rate limiting of DetNet traffic and the starvation avoiding of non-DetNet traffic, e.g., at the ingress of the DetNet domain must be applied by traffic policing and shaping functions. As [RFC2475] defined, the traffic conditioner may contain four elements: meter, marker, shaper and dropper. Traffic conditioning performs metering, shaping, policing and/or re-marking to ensure the traffic which entering the DiffServ domain conforms to the service provisioning policy.

In DetNet, the traffic policing and conditioning SHOULD include meter, marker, shaper, dropper, scheduler and order. A meter with a DetNet Profile is used to measure the DetNet flows selected by a DetNet classifier and the result of the meter with respect to a packet may be used to trigger a DetNet action including a marking, shaping, dropping, scheduling or ordering. A marker is used to set the Cos field of a DetNet packet to a DetNet DSCP (section 2.3), mapping the marked packet to a DetNet PHB. A Shaper may apply specific shaping algorithms implemented by DetNet network, e.g., credit-based shaper [IEEE802.1Qav]. A dropper is used to discard some of the non-DetNet packets to provide the QoS of the DetNet flows when congestion occurs.

2.2.1. Scheduler

As described in [RFC8655], the DetNet flows can be scheduled to achieve time-based synchronization for scheduled traffic. This document proposes a new type of action for DetNet traffic conditioning named Scheduler action. A scheduler may apply specific scheduling and related Queuing algorithms implemented by DetNet network, e.g., Time-gated queues [IEEE802.1Qbv] and Cyclic Queuing and Forwarding [IEEE802.1Qch].

2.2.2. Order

As defined in [I-D.ietf-detnet-mpls], DetNet control word (d-CW) containing sequencing information for packet replication and duplicate elimination purposes. Sequence Number is different packet-by-packet. Based on Detnet MPLS data plane encapsulation, this document proposes a new type of action for DetNet traffic conditioning named order action which used to reorder the packets within a DetNet flow that are received out of order.

2.3. DetNet DSCP

The DetNet DSCP carried in CoS field in IP header and TC field in MPLS header may be used to mark packets at ingress nodes and select a DetNet PHB (section 2.4) at transit nodes. DetNet DSCP MUST be defined to one or more particular values, which MUST be unique for codepoints in the standard space.

[Ed.note: We need to define one or more DetNet DSCP values and related DetNet PHB for DetNet-specific treatment.]

2.4. DetNet PHB

As specified in [RFC2475], per-hop behaviors are defined to permit a reasonably granular means of allocating buffer and bandwidth resources at each node among competing traffic streams. PHB groups will usually share a common constraint such as a packet scheduling or buffer management policy. According to [RFC4594], Default Forwarding (DF) PHB, Assured Forwarding (AF) PHBs, Expedited Forwarding (EF) PHB and Class Selector (CS) PHBs have been defined to provide forwarding treatment. These PHBs can be used to forward DetNet flows based on the requirement.

This document defines a new type of Deterministic Networking (DN) PHB which is intended for traffic requiring extremely low data loss rates and bounded latency for DetNet. The DN PHB may include a set of PHB classes, e.g., DN1, DN2, etc. And the number of the class is the same with the DetNet DSCP values. The DSCP in IP header and TC in MPLS

header should be mapped to DN PHB with the relevant PHB specification which may be completed in future discussion.

2.5. DetNet Queuing

As discussed in [RFC8655], the nodes in DetNet network shall queue each received packets to one of the potential transmission ports and provide storage for queued packets, awaiting to submit these for transmission. A port provides one or more queues corresponding to the number of traffic classes. The queuing mechanism should be configured and implemented to DetNet nodes.

As defined in [RFC4594], Priority Queuing (PQ) was defined to queue the packets in priority sequence and Rate Queuing (RQ) selects packets according to the specified rate including Weighted Fair Queuing (WFQ) and Weighted Round Robin (WRR). Active Queue Management (AQM) also be defined to use packet dropping or marking to manage the depth of a queue.

As per IEEE 802.1 WG, queuing and transmission selection algorithms also can be used for queue scheduling in DetNet network.

3. DetNet IP DiffServ Consideration

As specified in [I-D.ietf-detnet-ip], no DetNet specific encapsulation is defined to support DetNet IP flow identification and DetNet service delivery. So the DetNet IP classification is the same as defined in [RFC2474] and [RFC2475]. But the recommended DetNet DSCP may be used to mark packets to select a DetNet PHB and the transit nodes should implement mechanisms performing the PHB. The mapping of DSCP to PHBs MUST be configurable. Implementations should support the recommended codepoint-to-PHB mappings in their default configuration.

4. DetNet MPLS DiffServ Consideration

As defined in [I-D.ietf-detnet-mpls], DetNet S-Label and F-Labels can be used in combination with MPLS TC filed in MF classifier. The BA classifier is the same with the [RFC3270].

Two types of LSPs including Explicitly TC-encoded-PSC LSP (E-LSP) and Label-Only-Inferred-PSC LSP (L-LSP) follows the definition of [RFC3270] and can be used to support DetNet explicit routes in MPLS-TE LSP. A E-LSP can be used to support one or more DetNet flows and a L-LSP can be established for one flow. E-LSP and L-LSP can use a signaled TC->PHB mapping to forward packets whose corresponding PHBs are defined in this document.

In DetNet MPLS network, DetNet Layer Two Service is supported in TSN over MPLS. The LSP egressing over edge nodes can use the preconfigured PHB->802.1 mapping as defined in [RFC3270].

As specified in [RFC3270], there may be more than one LSP carrying the same flow. Two or more LSPs can be merged into one LSP at one egressing LSR. It can be used to perform the packet replication (PRF) at ingress nodes and the packet elimination (PEF) at the egress nodes in DetNet DiffServ model. The order action which defined in this document can be used for packet ordering functionality (POF).

5. Security Considerations

TBD.

6. IANA Considerations

TBD.

7. Acknowledgements

TBD.

8. References

8.1. Informative References

[RFC2475] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., and W. Weiss, "An Architecture for Differentiated Services", RFC 2475, DOI 10.17487/RFC2475, December 1998, <<https://www.rfc-editor.org/info/rfc2475>>.

8.2. Normative References

[I-D.ietf-detnet-ip]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: IP", draft-ietf-detnet-ip-03 (work in progress), October 2019.

[I-D.ietf-detnet-mpls]
Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: MPLS", draft-ietf-detnet-mpls-03 (work in progress), October 2019.

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC2474] Nichols, K., Blake, S., Baker, F., and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, DOI 10.17487/RFC2474, December 1998, <<https://www.rfc-editor.org/info/rfc2474>>.
- [RFC3270] Le Faucheur, F., Wu, L., Davie, B., Davari, S., Vaananen, P., Krishnan, R., Cheval, P., and J. Heinanen, "Multi-Protocol Label Switching (MPLS) Support of Differentiated Services", RFC 3270, DOI 10.17487/RFC3270, May 2002, <<https://www.rfc-editor.org/info/rfc3270>>.
- [RFC4594] Babiarz, J., Chan, K., and F. Baker, "Configuration Guidelines for DiffServ Service Classes", RFC 4594, DOI 10.17487/RFC4594, August 2006, <<https://www.rfc-editor.org/info/rfc4594>>.
- [RFC5462] Andersson, L. and R. Asati, "Multiprotocol Label Switching (MPLS) Label Stack Entry: "EXP" Field Renamed to "Traffic Class" Field", RFC 5462, DOI 10.17487/RFC5462, February 2009, <<https://www.rfc-editor.org/info/rfc5462>>.
- [RFC8655] Finn, N., Thubert, P., Varga, B., and J. Farkas, "Deterministic Networking Architecture", RFC 8655, DOI 10.17487/RFC8655, October 2019, <<https://www.rfc-editor.org/info/rfc8655>>.

Authors' Addresses

Quan Xiong
ZTE Corporation
No.6 Huashi Park Rd
Wuhan, Hubei 430223
China

Phone: +86 27 83531060
Email: xiong.quan@zte.com.cn

Jinghai Yu
ZTE Corporation
50 Software Avenue, YuHuaTai District
Nanjing, Jiangsu 210012
China

Phone: +86 025 26774049
Email: yu.jinghai@zte.com.cn

Peng Liu
China Mobile
Beijing 100053
China

Email: liupengyjy@chinamobile.com

Fengwei Qin
China Mobile
Beijing
China

Email: qinfengwei@chinamobile.com

DeNet WG
Internet-Draft
Intended status: Standards Track
Expires: May 4, 2020

Q. Xiong
Y. Han
ZTE Corporation
F. Qin
P. Liu
China Mobile
November 1, 2019

DetNet QoS Yang
draft-xiong-detnet-qos-yang-02

Abstract

This document defines a YANG data model for Deterministic Networking (DetNet) Quality of Service (QoS) based on the Differentiated Services (DiffServ) model.

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 May 4, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
1.1. Requirements Language	2
1.2. Terminology	2
2. DetNet DiffServ QoS Model	3
2.1. DetNet QoS Tree Structure	3
2.2. DetNet QoS Module	4
3. Security Considerations	13
4. IANA Considerations	13
5. Acknowledgements	13
6. References	13
6.1. Informative References	13
6.2. Normative References	13
Authors' Addresses	14

1. Introduction

Deterministic Networking (DetNet) as defined in [RFC8655], provides a capability to carry specified unicast or multicast data flows for real-time applications with extremely low data loss rates and bounded latency. In the meanwhile, DetNet and non-DetNet packets are allowed to be transmitted in the same network and more than one DetNet flows which has different priorities may be forwarded through the DetNet domain. As discussed in [I-D.ietf-detnet-ip] and [I-D.xiong-detnet-qos-policy], the Differentiated Services (DiffServ) can be used to provide Quality of Service (QoS) for DetNet services.

This document defines a YANG data model for DetNet QoS based on the DiffServ model.

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 [RFC2119].

1.2. Terminology

The terminology is defined as [RFC8655], [RFC3270], [RFC2474] and [RFC2475].

2. DetNet DiffServ QoS Model

This document defines a YANG data module for DetNet DiffServ QoS Model as discussed in [I-D.xiong-detnet-qos-policy]. In the ietf-detnet-qos module, this is performed as one of the DetNet QoS policy.

2.1. DetNet QoS Tree Structure

DetNet DiffServ model is one type of the DetNet QoS policy and other policy types can be defined in detnet-policy-type.

[I-D.xiong-detnet-qos-policy] specified two types of classifiers including BA (Behavior Aggregate) and MF (Multi-Field) classifiers in detnet-classifier-type. DetNet IP BA classifier selects packets based on the DiffServ Code Point (DSCP) and DetNet MPLS BA classifier is based on the MPLS Traffic Class (TC) field. DetNet IP MF classifier selects packets based on the value of a combination of source address, destination address, DSCP, protocol ID, source port and destination port numbers and DetNet MPLS MF classifier is based on the MPLS TC, service label (S-Label) field and forwarding labels (F-Labels) of the header.

[I-D.xiong-detnet-qos-policy] defined a DetNet (DN) Per Hop Behavior (PHB) for DetNet forward other than existing PHBs including AF, EF, CS, DF etc. The PHB class information description is as qos-phb-class shown.

[I-D.xiong-detnet-qos-policy] defined two new types of action for DetNet traffic conditioning named order and scheduler action. Other actions including meter, shaper, dropper and marker as the detnet-action-type shown.

```
module: ietf-detnet-qos
  +--rw detnet-qos-policies
    +--rw detnet-policy-template* [detnet-policy-name]
      +--rw detnet-policy-name      string
      +--rw detnet-policy-type?     detnet-policy-type
      +--rw detnet-classifier-template* [detnet-classifier-name]
        +--rw detnet-classifier-name  string
        +--rw detnet-classifier-type?  detnet-classifier-type
        +--rw (classifier-type)?
          +--:(ba)
            +--rw (encapsulation-type)?
              +--:(MPLS)
                +--rw mpls-ba* [tc-value]
                  +--rw phb-class?  qos-phb-class
                  +--rw tc-value    uint8
              +--:(IP)
```

```

    +--rw ip-ba* [dscp-value]
      +--rw phb-class?    qos-phb-class
      +--rw dscp-value    uint8
+--:(mf)
  +--rw (tunnel-type)?
    +--:(MPLS)
      +--rw mpls-mf* [tc-value]
        +--rw phb-class?    qos-phb-class
        +--rw tc-value      uint8
        +--rw s-label?      uint32
        +--rw f-labels* [f-label-id]
          +--rw f-label-id    uint32
    +--:(IPv4)
      +--rw ipv4-mf* [dscp-value]
        +--rw phb-class?    qos-phb-class
        +--rw dscp-value    uint8
        +--rw ipv4-source-address?    inet:ipv4-address
        +--rw ipv4-destination-address?    inet:ipv4-address
        +--rw protocol-ID?    uint8
        +--rw source-port-numbers?    inet:port-number
        +--rw destination-port-numbers?    inet:port-number
    +--:(IPv6)
      +--rw ipv6-mf* [dscp-value]
        +--rw phb-class?    qos-phb-class
        +--rw dscp-value    uint8
        +--rw ipv6-source-address?    inet:ipv6-address
        +--rw ipv6-destination-address?    inet:ipv6-address
        +--rw protocol-ID?    uint8
        +--rw source-port-numbers?    inet:port-number
        +--rw destination-port-numbers?    inet:port-number
        +--rw flow-label?    inet:ipv6-flow-label
+--rw detnet-action* [detnet-action-type]
  +--rw detnet-action-type    detnet-action-type
  +--rw (actions)?
    +--:(meter)
    +--:(marker)
    +--:(shaper)
    +--:(dropper)
    +--:(order)
    +--:(scheduler)

```

2.2. DetNet QoS Module

```

<CODE BEGINS> file "detnet-diffserv-qos@2018-10-13.yang"
module ietf-detnet-qos {
  yang-version 1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-detnet-qos";
  prefix detnet-qos;

```

```
    import ietf-inet-types{
    prefix "inet";
    }

    organization "IETF DetNet Working Group";
    contact
    "WG Web:    <http://tools.ietf.org/wg/detnet/>
    WG List:    <mailto:detnet@ietf.org>
    WG Chair:    Lou Berger
                <mailto:lberger@labn.net>
                Janos Farkas
                <janos.farkas@ericsson.com>
    Editor:    Quan Xiong
                <mailto:xiong.quan@zte.com.cn>
    Editor:    Yufang Han
                <mailto:han.yufang1@zte.com.cn>";

    description
    "This YANG module describes the Deterministic Networking (DetNet)
    Quality of Service (QoS) based on the Differentiated Services (DiffSe
rv)
        model.";

    revision "2018-10-13" {
        description "initial revision";
        reference "RFC XXXX: draft-xiong-detnet-qos-yang-01";
    }

    typedef qos-phb-class {
    type enumeration {
        enum df {
            value 1 ;
            description "Default Forwarding for Best effort";
        }
        enum af1 {
            value 2 ;
            description "Assured forwarding class 1";
        }
        enum af2 {
            value 3 ;
            description "Assured forwarding class 2";
        }
        enum af3 {
            value 4 ;
            description "Assured forwarding class 3";
        }
        enum af4 {
            value 5 ;
            description "Assured forwarding class 4";
```



```
    }
    enum ef {
        value 6 ;
        description "Expedited forward";
    }
    enum cs6 {
        value 7 ;
        description "Internetwork control service class";
    }
    enum cs7 {
        value 8 ;
        description "Network control service class";
    }
    enum dn {
        value 9 ;
        description "DetNet forward";
    }
}
description
    "The PHB class including AF,EF,CS,DF,DN";
}

typedef detnet-policy-type {
    type enumeration {
        enum diffserv {
            value 1 ;
            description "DiffServ Policy";
        }
    }
    description
        "The DetNet policy type.";
}

typedef detnet-classifier-type {
    type enumeration {
        enum ba {
            value 1 ;
            description "DiffServ BA Classifier";
        }
        enum mf {
            value 2 ;
            description "DiffServ MF Classifier";
        }
    }
    description
        "The DetNet classifier type including BA and MF.";
}
```

```
typedef detnet-action-type {
  type enumeration {
    enum meter {
      value 1 ;
      description "DiffServ meter Action";
    }
    enum shaper {
      value 2 ;
      description "DiffServ shaper Action";
    }
    enum dropper {
      value 3 ;
      description "DiffServ dropper Action";
    }
    enum marker {
      value 4 ;
      description "DiffServ marker Action";
    }
    enum order {
      value 5 ;
      description "DiffServ order Action";
    }
    enum scheduler {
      value 6 ;
      description "DiffServ scheduler Action";
    }
  }
  description
  "The DetNet classifier type including BA and MF.";
}

grouping mpls-tc {
  description "MPLS TC Information";
  leaf phb-class {
    type qos-phb-class;
    description "Specify phb class of PHB info, support [a"
      + "f1,af2,af3,af4,be,ef,cs6,cs7,dn]";
  }
  leaf tc-value {
    type uint8 {
      range 0..7{
        description "MPLS-TC value, support [0-7]";
      }
    }
    mandatory true ;
    description "MPLS-TC value, support [0-7]";
  }
}
```

```

grouping ip-dscp {
  description "IP DSCP Information";
  leaf phb-class {
    type qos-phb-class ;
    description "Specify server class of PHB info, support [a"
      + "f1,af2,af3,af4,be,ef,cs6,cs7,dn]";
  }
  leaf dscp-value {
    type uint8 {
      range 0..63 {
        description "IPv4/IPv6 DSCP value, support [0-63]";
      }
    }
    mandatory true ;
    description "IPv4/IPv6 DSCP value, support [0-63]";
  }
}

grouping mpls-header-info {
  description "MPLS TC Information";
  leaf phb-class {
    type qos-phb-class ;
    description "Specify phb class of PHB info, support [a"
      + "f1,af2,af3,af4,be,ef,cs6,cs7,dn]";
  }
  leaf tc-value {
    type uint8 {
      range 0..7 {
        description "MPLS-TC value, support [0-7]";
      }
    }
    mandatory true ;
    description "MPLS-TC value, support [0-7]";
  }
  leaf s-label {
    type uint32;
    description "DetNet Flow ID value, support classifier MF";
  }
  list f-labels {
    key "f-label-id";
    description "DetNet forwarding label id, support classif
ier MF";
    leaf f-label-id {
      type uint32;
      description "DetNet forwarding label value, supp
ort classifier MF";
    }
  }
}

```

```

grouping ipv4-header-info {
  description "IP DSCP Information";
  leaf phb-class {
    type qos-phb-class ;
    description "Specify server class of PHB info, support [a"
      + "f1,af2,af3,af4,be,ef,cs6,cs7,dn]";
  }
  leaf dscp-value {
    type uint8 {
      range 0..63 {
        description "IPv4/IPv6 DSCP value, support [0-63]";
      }
    }
    mandatory true ;
    description "IPv4/IPv6 DSCP value, support [0-63]";
  }
  leaf ipv4-source-address {
    type inet:ipv4-address;
    description "source address value, support classifier MF";
  }
  leaf ipv4-destination-address {
    type inet:ipv4-address;
    description "destination address value, support classifier M
F";
  }
  leaf protocol-ID {
    type uint8;
    description "protocol ID, support classifier MF";
  }
  leaf source-port-numbers {
    type inet:port-number;
    description "source port numbers, support classifier MF";
  }
  leaf destination-port-numbers {
    type inet:port-number;
    description "destination port numbers, support classifier MF
";
  }
}

grouping ipv6-header-info {
  description "IPv6 DSCP Information";
  leaf phb-class {
    type qos-phb-class ;
    description "Specify server class of PHB info, support [a"
      + "f1,af2,af3,af4,be,ef,cs6,cs7,dn]";
  }
  leaf dscp-value {
    type uint8 {
      range 0..63 {

```

```

        description "IPv4/IPv6 DSCP value, support [0-63]";
    }
    mandatory true ;
        description "IPv4/IPv6 DSCP value, support [0-63]";
    }
    leaf ipv6-source-address {
        type inet:ipv6-address;
        description "source address value, support classifier MF";
    }
    leaf ipv6-destination-address {
        type inet:ipv6-address;
        description "destination address value, support classifier M
F";
    }
    leaf protocol-ID {
        type uint8;
        description "protocol ID, support classifier MF";
    }
    leaf source-port-numbers {
        type inet:port-number;
        description "source port numbers, support classifier MF";
    }
    leaf destination-port-numbers {
        type inet:port-number;
        description "destination port numbers, support classifier MF
";
    }
    leaf flow-label {
        type inet:ipv6-flow-label;
        description
            "The flow label of the header.";
    }
}

grouping detnet-classifiers {
    description "Configure the DetNet classifiers";
    choice classifier-type {
        description "Choice of classifiers types";
        case ba {
            description "BA classifier";
            choice encapsulation-type {
                description "Tunnel type includes: IP, MPLS.";
                case MPLS {
                    list mpls-ba {
                        key "tc-value";
                        description "MPLS-TC be mapped to PH
B";
                    }
                    uses mpls-tc;
                }
            }
        }
    }
}

```

```

        case IP {
            list ip-ba {
                key "dscp-value";
                description "IPv4/IPv6 DSCP be mapped
d to PHB";
                uses ip-dscp;
            }
        }
        case mf {
            description "MF classifier";
            choice tunnel-type {
                description
                    "Tunnel type includes: IPv4, IPv6, MPLS.";
                case MPLS {
                    list mpls-mf {
                        key "tc-value";
                        description "MPLS-TC be mapped to PH
B";
                        uses mpls-header-info;
                    }
                }
                case IPv4 {
                    list ipv4-mf {
                        key "dscp-value";
                        description "IPv4 DSCP be mapped to
PHB";
                        uses ipv4-header-info;
                    }
                }
                case IPv6 {
                    list ipv6-mf {
                        key "dscp-value";
                        description "IPv6 DSCP be mapped to
PHB";
                        uses ipv6-header-info;
                    }
                }
            }
        }
    }

    grouping detnet-actions {
        description
            "DetNet Configuration about the actions";
        list detnet-action {
            key "detnet-action-type";
            description "DetNet actions, to be defined.";
            leaf detnet-action-type {
                type detnet-action-type;
            }
        }
    }

```

```
        description "DetNet action types";
    }
    choice actions {
        description "Choice of action types";
        case meter {
            description "meter action";
        }
        case marker {
            description "marker action";
        }
        case shaper {
            description "shaper action";
        }
        case dropper {
            description "dropper action";
        }
        case order {
            description "order action";
        }
        case scheduler {
            description "scheduler action";
        }
    }
}

container detnet-qos-policies {
    description "Configuration about DetNet QoS Policy";
    list detnet-policy-template {
        key detnet-policy-name;
        description "DetNet policy template";
        leaf detnet-policy-name {
            type string;
            description "DetNet policy name";
        }
        leaf detnet-policy-type {
            type detnet-policy-type;
            description "DetNet policy type";
        }
    }
    list detnet-classifier-template {
        key detnet-classifier-name;
        description "DetNet classifier template";
        leaf detnet-classifier-name {
            type string;
            description "DetNet classifier name";
        }
        leaf detnet-classifier-type {
            type detnet-classifier-type;
        }
    }
}
```

```

        description "DetNet classifier type";
    }
    uses detnet-classifiers;
    uses detnet-actions;
}
}
}
}
<CODE ENDS>
```

3. Security Considerations

TBD.

4. IANA Considerations

TBD.

5. Acknowledgements

TBD.

6. References

6.1. Informative References

[RFC2475] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., and W. Weiss, "An Architecture for Differentiated Services", RFC 2475, DOI 10.17487/RFC2475, December 1998, <<https://www.rfc-editor.org/info/rfc2475>>.

6.2. Normative References

[I-D.ietf-detnet-ip] Varga, B., Farkas, J., Berger, L., Fedyk, D., Malis, A., Bryant, S., and J. Korhonen, "DetNet Data Plane: IP", draft-ietf-detnet-ip-03 (work in progress), October 2019.

[I-D.xiong-detnet-qos-policy] Xiong, Q. and Y. jinghai, "DetNet QoS Policy", draft-xiong-detnet-qos-policy-01 (work in progress), March 2019.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.

- [RFC2474] Nichols, K., Blake, S., Baker, F., and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, DOI 10.17487/RFC2474, December 1998, <<https://www.rfc-editor.org/info/rfc2474>>.
- [RFC3270] Le Faucheur, F., Wu, L., Davie, B., Davari, S., Vaananen, P., Krishnan, R., Cheval, P., and J. Heinanen, "Multi-Protocol Label Switching (MPLS) Support of Differentiated Services", RFC 3270, DOI 10.17487/RFC3270, May 2002, <<https://www.rfc-editor.org/info/rfc3270>>.
- [RFC5462] Andersson, L. and R. Asati, "Multiprotocol Label Switching (MPLS) Label Stack Entry: "EXP" Field Renamed to "Traffic Class" Field", RFC 5462, DOI 10.17487/RFC5462, February 2009, <<https://www.rfc-editor.org/info/rfc5462>>.
- [RFC8655] Finn, N., Thubert, P., Varga, B., and J. Farkas, "Deterministic Networking Architecture", RFC 8655, DOI 10.17487/RFC8655, October 2019, <<https://www.rfc-editor.org/info/rfc8655>>.

Authors' Addresses

Quan Xiong
ZTE Corporation
No.6 Huashi Park Rd
Wuhan, Hubei 430223
China

Phone: +86 27 83531060
Email: xiong.quan@zte.com.cn

Yufang Han
ZTE Corporation
50 Software Avenue, YuHuaTai District
Nanjing, Jiangsu 210012
China

Phone: +86 15951984307
Email: han.yufang1@zte.com.cn

Fengwei Qin
China Mobile
Beijing
China

Email: qinfengwei@chinamobile.com

Peng Liu
China Mobile
Beijing 100053
China

Email: liupengyjy@chinamobile.com