Operations and Management Area Working Group
Internet-Draft
Intended status: Informational
Expires: January 5, 2015

Q. Wu
M. Wexler
Huawei
M. Boucadair
France Telecom
S. Aldrin
Huawei USA
G. Mirsky
Ericsson
P. Jain
Nuage Networks

July 4, 2014

Problem Statement and Architecture for Transport-Independent Multiple Layer OAM
draft-ww-opsawg-multi-layer-oam-02.txt

Abstract

Operations, Administration, and Maintenance (OAM) mechanisms are critical building blocks in network operations that are used for service assurance, fulfillment, or service diagnosis, troubleshooting, and repair. The current practice is that many technologies rely on their own OAM protocols that are exclusive to a given layer. There is little consolidation of OAM in either data plane or management plane nor well-documented inter-layer OAM operations. Vendors and Operators dedicate significant resources and effort through the whole OAM life-cycle each time when a new technology is (to be) introduced. This is even exacerbated when dealing with integration of OAM across multiple technologies.

This document describes the problem space and defines an architecture for the generic and integrated OAM with a focus of multi-layer and cross-layer considerations.

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 5, 2015.

Table of Contents

1. Introduction

Operations, Administration, and Maintenance (OAM) mechanisms being understood and used in context of RFC 6291 [RFC6291] are critical building blocks in network operations that are used for service assurance, fulfillment, or service diagnosis, troubleshooting, and repair. The key foundations of OAM and its functional roles in monitoring and diagnosing the behavior of networks have been studied at OSI layers 1, 2 and 3 since a while. As a reminder, OAM functions are used in many management applications for various objectives such as (i) failure detection, (ii) reporting the defect/ failure information, (iii) defect/failure localization, (iv) performance monitoring, and (v) service recovery.

The current practice that consists in enabling OAM techniques for each layer has shown its limits; this is a need for cross-layer and inter-layer OAM considerations [RFC7276]. This need for inter-layer OAM is motivated by the need to achieve: network optimization, efficient enforcement of TE (Traffic Engineering) techniques including ensuring path diversity at distinct layers or computing completely disjoint paths at several layers, fine-grain tweaking, ease of root cause analysis, ability to maintain a network-wise visibility in addition to layer-specific one, etc.

It is worth to mention also that there are two restrictions for multi-layer structure as discussed in [RFC7276]:

- Each layer has its own OAM protocol, OAM should not cross layer boundaries.

- Each layer OAM used at different level of hierarchy in the network.

Moreover, there is little consolidation of OAM in either data plane or management plane. Vendors and operators dedicate a lot resources and effort through the whole OAM life-cycle each time a new technology is (to be) introduced. Integration of OAM across multiple

technologies in either data plane or management plane is extremely difficult to achieve.

When operating networks with more than one technology, maintenance and troubleshooting are achieved per technology and per layer, operation process can be very cumbersome since OAM is not defined to cross layer boundaries. Another challenge is presented by use of different technologies and corresponding OAM on the same layer of adjacent network domains. Interworking between different OAM often not defined and are left to proprietary solutions. In many cases when keeping network complexity down and simplifying OAM is needed, it is desirable to have a generic and integrated OAM to cover heterogeneous networking technologies.

This document defines the problem space and describes an architecture for the generic and integrated OAM in the multi-layer and multi-domain networks. In particular, it outlines the problems encountered with existing OAM protocols and their impact on introduction of new technologies (see Section 3).

This document covers the following:

- Data plane OAM consolidation by looking at the common active OAM functions (including, Connectivity Verification (CV), Path Verification and Continuity Checks (CC), Path Discovery, Performance Measurement) necessary to monitor and diagnose a network;

- Management plane consolidation by interacting with data plane OAM and abstracting OAM information common to different layer via uniformed interface.

2. Terminology

This document defines the following terms:

Transport Independent Multi-Layer OAM:

In an multi-layer network, transport independent OAM is OAM that can be deployed independent of media, data protocols, and routing protocols It denotes the ability to exchange OAM information across layers and domains between nodes along forwarding path, and gather OAM information that are common to different layers and expose it to the management application through a unified interface. These aspects are not specific to a given transport technology.

OAM function:

Refers to the atomic building blocks of OAM; an OAM function defines an OAM capability (See section 2.2.3 of [RFC7276]).

OAM protocol:

Refers to a protocol used for implementing one or more OAM functions (See section 2.2.3 of [RFC7276]).

OAM tool:

Denotes a specific means of applying one or more OAM functions. An OAM protocol can be an OAM tool. An OAM tool can use a set of OAM protocols or a set of protocols that are not strictly OAM related (See section 2.2.3 of [RFC7276]).

OAM packet:

Refers to a packet generated at Maintenance Point using an OAM protocol. An OAM packet, which carries OAM information, is usually forwarded through the same route/path as the data traffic and receive the same (forwarding) treatment.

Maintenance Domain (MD):

Refers to the part of a network where OAM function is performed (initiated).

Maintenance Point (MP):

Is a generic functional entity that is associated with a particular MD, defined at a specific layer of a network and can initiate and/or react to OAM packets.

Maintenance Endpoint (MEP):

Is an endpoint MP that initiates OAM packets and responds to them.

Maintenance Intermediary Point(MIP):

In between MEPs, there are zero or more intermediate points, called Maintenance Intermediary Point. A Maintenance Intermediary Point (MIP) is an intermediate MP that does not generally initiate OAM packets but is able to respond to OAM packets that are destined to it.

Maintenance Association (MA):

The relationship between a set of MEPs to which maintenance and monitoring operations apply.

Network Element (NE):

Denotes a physical or virtual network device/function that connects directly to the network. NE can host MPs and provide network connectivity to one or many MPs.

2.1. Acronyms and Abbreviations

CC - Continuity Check

CV - Connectivity Verification

SNMP - Simple Network Management Protocol

NETCONF - Network Configuration

ETH - Ethernet

APS - Automatic Protection Switching

LT - LinkTrace

RDI - Remote Defect Indication

AIS - Alarm indication Signal

OWAMP - One Way Active Measurement Protocol

TWAMP - Two Way Active Measurement Protocol

CFM - Connectivity Fault Management

3. Problem Statement

OAM mechanisms are usually oriented toward a single network technology or a single layer. Each technology or layer has its best suited OAM tools. Some of them providing rich functionality rely on the capabilities of one protocol, while the others provide each function with a different protocol; In the current situation, there is little, or no re-use, of software and hardware for each OAM protocol.

Integration of OAM across multiple technologies is extremely difficult. Vendors and operators waste a lot through the whole OAM life-cycle when a new technology is introduced:

(1) Design and development: For every new protocol there is a need to invest in complete life-cycle (i.e.,the design and development of data, control and management planes). In some cases, even adding a single OAM function requires the above complete life-cycle.

(2) Operation and Maintenance: There is a need to re-train operation people for almost every newly introduced technology or feature. The above causes a slow time-to-market and a waste of time and effort for any new technology and/or OAM function.

Specifically, in Service Function Chaining environment, every Service Function may operate at a different layer and may use different encapsulation and tunneling techniques. When taking into account virtualization related technologies, the number of encapsulation and tunneling options increase even more. Still, end-to-end service OAM mechanisms and information exchanges between Service Functions should be provided to operate and maintain the network as a whole. This requires a generic toolkit that can provide all necessary tools in context of multi-technology, multi-layer, physical and virtual environments.

A particular problem is how OAM information at different layer is made available to a management application for use and learnt via the unified management interface. For example, in the case of an multi-layer network, OAM information needs to be imposed to the packet and injected into the network and at last abstracted from various layers and expose them to the management application.

3.1. Use of Existing Protocols

OAM information resides at each layer and may currently be exchanged at each network layer in a domain by using various encapsulation technologies at the Layer 2 & Layer 3 levels. OAM information may be gathered and exported from a domain (for example, northbound) using SNMP [RFC3411]or NETCONF/YANG [RFC6241].

It is desirable that a solution to the problem described in this document does not require the implementation of a new, network-wide protocol or introduce a shim layer to carry OAM information. Instead, it would be advantageous to make use of an existing protocols or functionalities that are commonly implemented and are currently deployed in operational networks. This has many benefits in network stability, time to deployment, and operator training.

It is recognized, however, that existing protocols or functionalities are unlikely to be immediately suitable to this problem space without some protocol extensions. Extending protocols must be done with care

and with consideration for the stability of existing deployments. In extreme cases, when there is a lack of functionality, although similar mechanisms exist in other technologies, a new protocol can be preferable to a "messy" hack of an existing protocol.

3.2. Strong Technology dependency

OAM protocols are relying heavily on the specific network technology they are associated with. For example, ICMP, LSP Ping are using different network technologies but provide the same OAM functionality, i.e., Path Discovery. Another example is BFD,LSP Ping are using different network technologies but provide the same functionality, i.e., Continuity Verification. Figure 1 shows common OAM functionalities shared by various existing IETF OAM protocols.

	Continuity Check	Connectivity Verification	Path Discovery	Performance Measurement
ICMP	Echo(Ping)		Traceroute	-Delay -Loss rough measurement
BFD	BFD Control /Echo	BFD Control		
LSP Ping		Ping	Traceroute	- Delay - Packet Loss
IPPM				-OWAMP -TWAMP
MPLS-TP OAM	CC (use of BFD)	CV (use of BFD or LSP Ping)	Traceroute	-Delay -Packet Loss

Figure 1: Examples of IETF OAM tools

3.3. Weakness of Cross-Layer OAM

Troubleshooting is cumbersome due to protocol variety and lack of multi-layer OAM. Usually OAM messages should not cross layer boundaries. Each of the service, network and transport layers

possesses its well-discernible and native OAM stream. In addition, OAM messages should not be leaked outside of a management domain within a layer, where a management domain is governed by a single business organization. When having networks with more than one technology, maintenance and troubleshooting are done per technology and layer.

These rules could in some cases ease the understanding in which technology the operation is done or fault is located. In some cases, when one layer OAM fails, it may be desirable to drop down to the another layer OAM and issue the corresponding OAM command, using the same APIs, if OAM in multiple layers can be supported. However, in most cases switching tools and layers in the same operation process is cumbersome and not serving the main idea - to find the root cause location. It would be very helpful to have a generic mechanisms that is end to end basis, allow management application interact with data plane OAM and can ping IPv4 host by an IPv6 source or having one tool to troubleshoot combined IP, MPLS, Ethernet, GRE and VXLAN network.

In Service Function Chaining environment, it is necessary to provide end-to-end OAM across certain or all entities and involving many layers. Inter-layer OAM considerations are key in an SFC context because problems may occur at the network layer or at the service chaining layer.

3.4. Lack of OAM above Layer 3

The Layer 2/3 OAM protocols are quite rich in their functionality, well defined, standardized and heavily used. In the last years a lot of work was conducted to consider maintenance domains and levels in order to better handle the issues of technology re-use, smooth interoperability and interworking between domains.

The above mechanisms are not defined for the technologies above Layer 3. Therefore, in the SFC environment where a Service Function Chaining is composed by a set of Service Functions, but providing an end-to-end chain or path from a source to destination in a given order [I.D-ietf-sfc-problem-statement], no standard exists as a reference for OAM since when the service packets is steered through a set of service nodes distributed in the network, each service node may act at different layers above layer 3.

3.5. Issues of Abstraction

In multi-layer network, OAM functions are enabled at different layers and various OAM information needs to be gathered from various layers. Without multi-layer OAM in place, it is hard for management applications to understand what information at different layers

stands for. One possible solution to these issues is to abstract the OAM information shared across layers, i.e., using the same tool or API to activate the OAM functions at different layers and retrieve the results.

The challenge is to abstract in a way that retains as much useful information as possible while filtering the data that is not needed to be leaked to other layers. An important part of this effort is a clear understanding of what information is actually needed.

3.6. Issue of OAM Information Gathering from Layers Covering Heterogeneous Network Technologies

In SFC, the service packets are steered through a set of service nodes (virtual or physical) hosting the service function distributed in the network. In the NVO3 network, the data packet may also traverse a set of overlay nodes distributed in the network. Overlay technologies or other tunneling technologies can be used to stitch these service nodes or overlay node in order to form end to end path.

When any overlay Segment or segment of service chain in the network fails to deliver user traffic, there is a need to provide a tool that would enable users to detect such failures at different layer using various encapsulation protocols and locate faults in the specific part of the network, and a mechanism to isolate these faults. It may also be desirable to test the data path before mapping user traffic to the Overlay Segment or segment of service chain. When multiple layer OAMs are used in the different parts of the network; how these layers OAM interwork at the boundary of each part of network is also a serious issue.

3.6.1. Focus on Service Function Chaining

When the service packets are steered through a set of Service Nodes (virtual or physical) hosting the Service Function distributed in the network, each Service Node may work at different layer above layer 3 and may embed several SFs. When OAM mechanism is applied, it is necessary to allow OAM packets to be exchanged:

- between Service Functions/Service Nodes and the SFC Management System,
- between these Service Nodes,
- between Service Functions at different layers,
- or between Service Nodes and ingress node of the SFC-enabled domain.

When Service Functions that are part of the SFC-enabled domain do support the OAM capability (e.g., an SFC-unaware Service Function) and Service Node has OAM capability, Service Nodes may be responsible for monitoring and diagnosing and reporting service availability of these Service Functions. It is more desirable to allow Service Functions register with a Service Node. Either Service Functions report status to the Service Node or the Service Node performs liveness check of the Service Function.

In addition, some Service Functions may not have Layer 2-3 switching/ routing capability and therefore are not aware of any OAM function at Layer 2-3. Also when there are no OAM functions at service Layers above layer 3, it is hard to identify the layer that can be used to gather OAM information when it comes to a fault situation or degradation of performance. For example, when a data packet is transmitted from SFC ingress node (i.e.,Classifier) and traverse a set of Service Nodes that host Service Function,the data packet may be discarded either at the SFC ingress node, one specific Service Node or one specific Service Function. Also the data packet may be lost between SFC ingress and one Service Node, or between two Service Nodes, or between one Service Node and one Service Function, how to detect the fault between them and how to isolate problem to that layer?

Editor's Note: Section 3.6.1 is too specific. This text can be presented as an example to illustrate a problem not a problem per se or moved to a use case draft.

4. Architecture Overview

Figure 2 shows the reference architecture for Layering OAM. This reference architecture assumes that

- Any network element can use different technologies and corresponding OAM on the same layer at the boundary of two adjacent domains
- Any two network element may provide service delivery at different layer
- Management entity can manage network devices in more than one maintenance domains.

In this architecture, three layers are defined:

M1: "Data Plane layer"

M2: "Management Plane layer"

M3: "Service Plane layer"

In the M1 layer, a typical network can be partitioned into several domains. Each domain has at least two MEPs and none or several MIPs. One domain can contain one or more maintenance associations (MAs). MEP is a maintenance functional entity that is implemented into a Network Element at the maintenance domain boundary and can send and receive OAM packets. MIP is a maintenance functional entity that is implemented into a Network Element in the maintenance domain and can forward OAM packets and respond OAM packets only when triggered by a specific OAM function (e.g., Path Discovery or Connectivity Verification). MEPs and MIPs can exist in the same maintenance domain and belong to different MAs. They can also exist at different layers and use various encapsulating protocols.

The M2 contains the interface which management entity uses to manage individual network devices. In this document, we further require management entities to use this interface as uniform interface (API and or UI) to gather OAM information from MEP and MIP in the network devices(either physical or virtual entity) and execute transactions or operations on MEP and MIP across domains, layers and vendors. Protocols that can be used to manipulate the configuration of a network device include SNMP [RFC1157], Command Line Interfaces, NETCONF [RFC6241], and other protocols.

On the M3 layer, there is a uniform interface (API and/or UI) that covers all the managed devices and can execute network-wide transactions. This layer allows applications and operators to execute configuration, monitoring and action tasks across multiple network devices, from a mix of domains, layers, vendors. Still the abstraction level is that of the network elements themselves, so whatever configuration, status, actions and notifications they provide, that is what you get here, but without having to worry about the location and the protocol to reach the device.

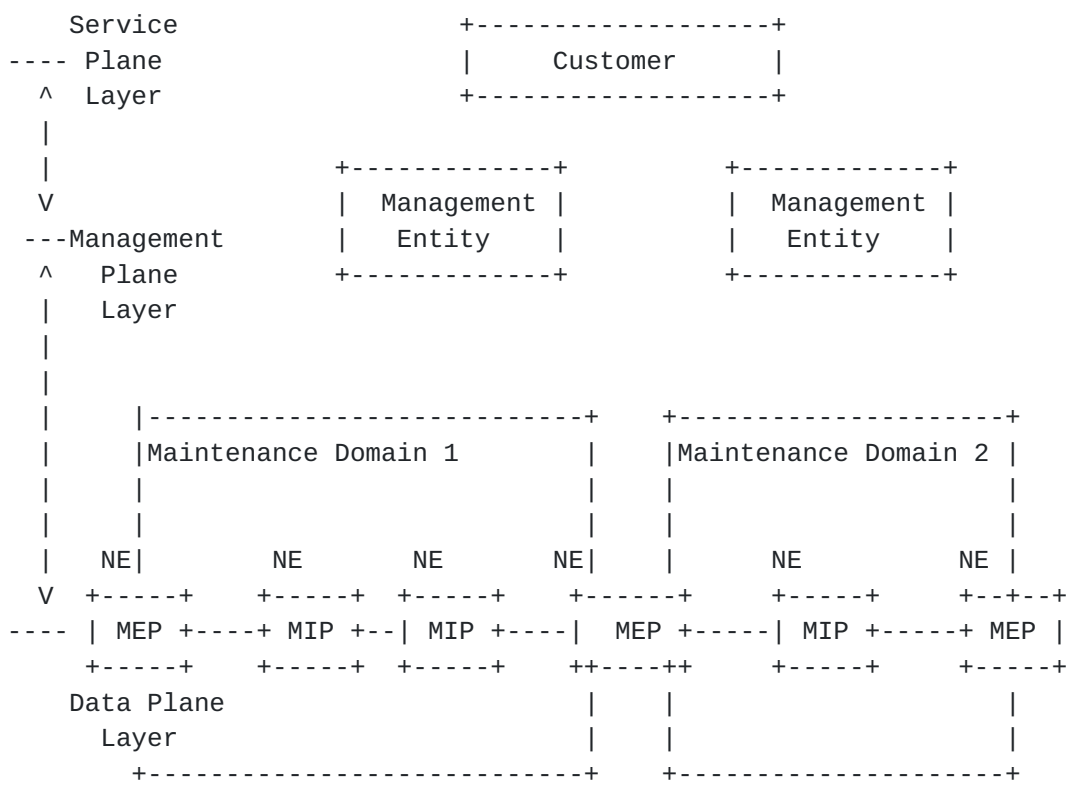

Figure 2: Architecture for Layering OAM in the management plane

An example of service-specific that depicts OAM layers can be found in [RFC4176] (L3VPN case).

5. Existing Work

The following discuss related IETF work and is provided for reference. This section is not exhaustive, rather it provides an overview of few initiatives focusing on the pain-points of OAM:

1. [I-D.tissa-netmod-oam] is an important work that creates a YANG unified data model for OAM that is based on IEEE CFM model. This model may be used also for IP OAM functionality. This effort is focused on the management plane of OAM and should be complemented by an accompanying data-plane and/or control-plane work. It may require also some extensions to address wider variety of functions and technologies.

2. Several contributions conducted in the past years, had tried to address new technologies using existing mechanisms. [I-D.jain-nvo3-overlay-oam] and MPLS-TP OAM documents are only examples for such efforts.

6. Architectural Consideration

6.1. Basic Components

6.1.1. Overlay OAM

6.1.2. OAM at the top of Layer 3

6.2. OAM Functions in the Data Plane

Many OAM functions may require protocol extensions or new protocol development to meet the transport requirements. In the existing OAM tools, Some of them providing rich functionality in one protocol,the other providing each function with a different protocol and each technology is developed independently.

To consolidate OAM in the data plane, the OAM in multi-layer Environment is expect to support the following common OAM functions used in OAM-related standards. These functions are used as building blocks in the data plane OAM standards described in this document.

6.2.1. Continuity Check

This type of mechanisms check that the monitored layer and/or entity are alive and providing path from specific point(s) to other point(s). Some examples are IP Ping, BFD [RFC5880] and ETH CC.

6.2.2. Connectivity Verification

Verifying that the actual connection is consistent with the required connection and no mis-connection occurred. Some examples are IP Ping, and ETH loopback.

6.2.3. Path Discovery

Used to discover the path that specific service traverses in the network. Some examples are LSP Traceroute, IP Traceroute and ETH-LT/ linktrace.

6.2.4. Performance Measurement

A function that monitors the performance parameters of a network entity. Such parameters could be Delay, Delay-variation, loss, availability of services and class of services. Examples are TWAMP[RFC5357]/ OWAMP[RFC4656] and Y.1731,MPLS Loss and Delay Measurement [RFC6374].

6.2.5. Protection Switching Coordination

A function that is used to signal protection switching states and commands. Examples are ETH APS messages and MPLS-TP Protection Switching Coordination OAM [RFC6378].

6.2.6. Alarm/defect Indication

A function that is used to indicate that a failure occurred downstream or upstream within a connection/service. Used also to trigger fast protection or to suppress alarms. Examples are ETH AIS and ETH RDI, MPLS-TP RDI [RFC6428].

6.2.7. Maintenance Commands

A function that is used to signal a maintenance state or command within a connection/service. Examples can be ETH Lockout.

6.3. OAM in Management Plane

Management systems play an important role in configuring or provisioning OAM functionality consistently across all devices in the network, and for automating the monitoring and troubleshooting of network faults. However OAM is not provision. In general, provisioning is used to configure the network to provide new services, whereas OAM is used to keep the network in a state that it can support already existing services.

As we know each layer has its own OAM protocols. OAM can be used at different levels of hierarchy in the network to form a multi-layer OAM solution [RFC7276]. To support multi-layer OAM covering various heterogeneous transport technologies, the OAM in the management needs to be consolidated as follows:

- OAM information needs to be abstracted that are common to different layer and different domain.
- Support customized OAM service, e.g., customized service diagnose.
- OAM information is provided to management entity from managed device via a uniform interface (API and/or UI)
- Sets up MD MEP and MIP in the network provision phase
- Enables basic OAM functionality(e.g., enable the origin of ping and trace packets or configure Connectivity Fault Management (CFM)) on the managed devices in the service activation phase.

The different OAM tools may be used in one of two basic types of activation:

- Proactive activation - indicates that the tool is activated on a continual basis, where messages are sent periodically, and errors are detected when a certain number of expected messages are not received.

- On-demand activation - indicates that the tool is activated "manually" to detect a specific anomaly.

7. Building on Existing Protocols

8. Scoping Future Work

This section includes a set of candidate items for activities to be conducted within IETF.

These objectives are not frozen; further discussion is required to target key issues and scope the work to be conducted within IETF accordingly.

Candidate investigation items are listed below:

- Understand and discuss situations where an OAM protocol can be tuned and optimized for a specific data plane.

- OAM consolidation in the data plane:

  * Exchange OAM information at the service layer atop of layer 3.

  * Deployed over various encapsulating protocols, and in various medium types

- OAM consolidation in the management plane:

  * Abstract OAM information common to different layers.

  * Expose OAM information via unified interface to management entities, independently of the layer they belong to.

  * Discuss how information gathered from various layers can be correlated for the sake of network operations optimization purposes.

  * Propose means to help during service diagnosis; these means may rely on filtering information to be leaked to other layers so that time recovery can be optimized. A typical example would

be efficient root cause analysis that is fed with input from various layers.

* Propose means that would help to optimize a network as a whole instead of the monolithic approach that is specific to a given layer. For example, investigate means that would help in computing diverse and completely disjoint paths, not only at layer 3 but also at the physical layer.

9. Manageability Considerations

10. Security Considerations

Security considerations are not addressed in this problem statement only document. Given the scope of OAM, and the implications on data and control planes, security considerations are clearly important and will be addressed in the specific protocol and deployment documents.

11. Acknowledgements

The authors would like to thank Romascanu, Dan, Tom Taylor, Tissa Senevirathne,Huub van Helvoort, Yuji Tochio for their valuable reviews and suggestions.

12. References

12.1. Normative References

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

[RFC6291] Andersson, L., Helvoort, H., Bonica, R., Romascanu, D., and S. Mansfield, "Guidelines for the Use of the "OAM" Acronym in the IETF", RFC 6291, June 2011.

[RFC7276] Mizrahi, T. and N. Sprecher, "An Overview of Operations, Administration, and Maintenance (OAM) Tools", RFC 7276, June 2014.

12.2. Informative References

[I-D.jain-nvo3-overlay-oam] Jain, P., "Generic Overlay OAM and Datapath Failure Detection", ID draft-jain-nvo3-overlay-oam-01, February 2014.

[I-D.tissa-netmod-oam]
Senevirathne , T., Finn, N., Kumar , D., and S. Salam , "YANG Data Model for Operations Administration and Maintenance (OAM)", ID draft-tissa-netmod-oam-00, March 2014.

[I.D-ietf-sfc-problem-statement]
Quinn, P., Guichard, J., and S. Surendra, "Network Service Chaining Problem Statement", ID draft-ietf-sfc-problem-statement, August 2013.

[RFC3411] Harrington, D. and R. Presuhn, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks", RFC 3411, December 2002.

[RFC4176] El Mghazli, Y., Nadeau, T., Boucadair, M., Chan, K., and A. Gonguet, "Framework for Layer 3 Virtual Private Networks (L3VPN) Operations and Management", RFC 4176, October 2005.

[RFC4656] Shalunov, S., Karp, A., Boote, J., and M. Zekauskas, "A One-way Active Measurement Protocol (OWAMP)", RFC 4656, September 2006.

[RFC5357] Hedeyat, K., Krzanowski, R., Morton, A., Yum, K., and J. Babiarz, "A Two-Way Active Measurement Protocol (TWAMP)", RFC 5357, October 2008.

[RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, June 2010.

[RFC6241] Enns, R., Bjorklund, M., Schoenwaelder, J., and A. Bierman, "Network Configuration Protocol (NETCONF)", RFC 6241, June 2011.

[RFC6374] Frost, D. and S. Bryant, "Packet Loss and Delay Measurement for MPLS Networks", RFC 6374, September 2011.

[RFC6378] Weingarten, Y., Bryant, S., Osborne, E., Sprecher, N., and A. Fuligoli, "Packet Loss and Delay Measurement for MPLS Networks", RFC 6378, October 2011.

[RFC6428] Allan, D., Swallow, G., and J. Drake, "Proactive Connectivity Verification, Continuity Check, and Remote Defect Indication for the MPLS Transport Profile", RFC 6428, November 2011.

Authors' Addresses

Qin Wu
Huawei
101 Software Avenue, Yuhua District
Nanjing, Jiangsu 210012
China

Email: bill.wu@huawei.com

Mishael Wexler
Huawei
Riesstr. 25
Munich 80992
Germany

Email: mishael.wexler@huawei.com

Mohamed Boucadair
France Telecom
Rennes 35000
France

Email: mohamed.boucadair@orange.com

Sam Aldrin
Huawei Technologies USA
2330 Central Expressway
NSanta Clara, CA 95051
USA

Email: aldrin.ietf@gmail.com

Greg Mirsky
Ericsson

Email: gregory.mirsky@ericsson.com

Pradeep Jain
Nuage Networks
755 Ravendale Drive
Mountain View, CA 94043
USA

Email: pradeep@nuagenetworks.net