Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 31, 2013

A. Akhter
Cisco Systems
H. Scholz
VOIPFUTURE GmbH
July 30, 2012

IPFIX Information Elements for RTP Flow Performance Measurement
draft-akhter-opsawg-perfmon-ipfix-03.txt

Abstract

There is a need to be able to quantify and report the performance of RTP based applications. This performance data provides information essential in validating service level agreements, fault isolation as well as early warnings of greater problems. This document describes IPFIX Information Elements related to RTP performance measurement of network based applications. In addition, to the performance information several non-metric information elements are also included to provide greater context to the reports. The measurements use audio/video applications as a base but are not restricted to these class of applications. These new IPFIX Information Elements can describe the entire duration of an RTP stream or a smaller time slice of it.

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 http://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 31, 2013.

Copyright Notice

Table of Contents

1. Introduction

Today's networks support a multitude of highly demanding and sensitive network applications. Network issues are readily apparent by the users of these applications due to the sensitivity of these applications to impaired network conditions. Examples of these network applications include applications making use of IP based audio, video, database transactions, virtual desktop interface (VDI), online gaming, cloud services and many more. In some cases, the impaired application translates directly to loss of revenue. In other cases, there may be regulatory or contractual service level agreements that motivate the network operator. Due to the sensitivity of these types of applications to impaired service it leaves a poor impression of the service on the user-- regardless of the actual performance of the network itself. In the case of an actual problem within the network service, monitoring the performance may yield a early indicator of a much more serious problem.

Due to the demanding and sensitive nature of these applications, network operators have tried to engineer their networks towards wringing better and differentiated performance. However, that same differentiated design prevents network operators from extrapolating observational data from one application to another, or from one set of synthetic (active test) test traffic to actual application performance. This gap highlights the importance of generic measurements as well as the reliance on user traffic measurements-- rather than synthetic tests.

Performance measurements on user data provide greater visibility not only into the quality of experience of the end users but also visibility into network health. With regards to network health, as flow performance is being measured, there will be visibility into the end to end performance which means that not only visibility into local network health, but also viability into remote network health. If these measurements are made at multiple points within the network (or between the network and end device) then there is not only identification that there might be an issue, but a span of area can be established where the issue might be. The resolution of the fault increases with the number of measurement points along the flow path.

IP based applications, esp. those with real-time requirements, may suffer temporarily from impairments such as bandwidth bottlenecks or packet loss. Performance measurement with average values is not able to record and highlight these issues. Due to this the measurement interval must be configurable to a short time slice in order to indicate such temporary impairments. Aggregation of measurements shall be possible to aggregate multiple measurements of the same application stream or multiple streams of the same type but

potentially generated by different users.

The IP Flow Information Export Protocol (IPFIX) [RFC5101] provides new levels of flexibility in reporting from measurement points across the life cycle of a network based application. IPFIX can provide granular results in terms of flow specificity as well as time granularity. At the same time, IPFIX allows for summarization of data along different types of boundaries for operators that are unconcerned about specific sessions but about health of a service or a portion of the network. This document details the expression of IPFIX Information Elements whose calculation is defined in an accompanying document.

As this document covers the reporting of these metrics via IPFIX, consideration is taken with mapping the metric's capabilities and context with the IPFIX information and data representation model. The guidelines outlined in [I-D.trammell-ipfix-ie-doctors] are used to ensure proper IPFIX information element definition.

There has been related work in this area such as [RFC2321]. [I-D.huici-ipfix-sipfix], and [VoIP-monitor].

2. Terminology

Terms used in this document that are defined in the Terminology section of the IPFIX Protocol [RFC5101] document are to be interpreted as defined there.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

In addition, the information element definitions use the following terms:

Name: Name of the information element per the IPFIX rules defined in Section 2.3 of [RFC5102]

Description: Short description of what the information element is trying to convey.

Observation Point: Where the measurement is meant to be performed. Either at an intermediate point (for example, a router) or end system.

Element Data Type: The IPFIX informationElementDataType as defined in Section 3.1 of [RFC5610]

Element Semantics: The IPFIX informationElementSemantics as defined in section Section 3.6 of [RFC5610]

Element Units: The IPFIX informationElementUnits as defined in section Section 3.7 of [RFC5610]

Element Range Begin: The IPFIX informationElementRangeBegin as defined in section Section 3.7 of [RFC5610]

Element Range End: The IPFIX informationElementRangeEnd as defined in section Section 3.7 of [RFC5610]

Element Id: The IPFIX global unique element ID as defined in Section 3.2 of [RFC5101]

Status: The status of the specification of this IPFIX Information Element.

3. General Usage

3.1. Quality of Service (QoS) Monitoring

For QoS monitoring, it is important to be able to capture the application context. For example, in the case of interactive audio flows, the codec and the fact that the application is interactive should be captured. The codec type can be used to determine loss thresholds affecting end user quality and the interactive nature would suggest thresholds over one way delay. The IPFIX reporting would need to keep this information organized together for operator to be able to perform correlated analysis.

3.2. Fault Isolation and Troubleshooting

It has been generally easier to troubleshoot and fix problems that are binary in nature: it either works or does not work. The host is pingable or not pingable. However, the much more difficult to resolve issues that are transitory in nature, move from location to location, more complicated that simple ICMP reachability and many times unverifiable reports by the users themselves. It is these intermittent and seemingly inconsistent network impairments that performance metrics can be extremely helpful with. Just the basic timely detection that there is a problem (or an impending problem) can give the provider the confidence that there is a real problem that needs to be resolved. The next step would be to assist the

operator in a speedy resolution by providing information regarding the network location and nature of the problem.

Transient problems which affect a user only for a short time of this session can be made visible with measurements taken in short fixed time slices, e.g. every 10 seconds. While a traditional measurement on a per session basis may not show an intermittent impairment (e.g. packet loss) the short measurement interval highlights these.

4. New Information Elements

The information elements are organized into two main groups:

Transport Layer: Metrics that might be calculated from observations at higher layers but essentially provide information about the network transport of user date. For example, the metrics related to packet loss, latency and jitter would be defined here.

RTP Header: Information Elements that describe the RTP stream properties based on RTP header information but not the RTP payload itself.

RTP Payload: Information Elements that describe the RTP payload. For example, packet count and media type.

User and Application Layer: Metrics that are might be affected by the network indirectly, but are ultimately related to user, end-system and session states. For example, session setup time, transaction rate and session duration would be defined here.

4.1. Transport Layer

4.1.1. perfObservationType

Name: perfObservationType

Description: The observation type is analog to sipObservationType defined in [trammel sip-msg]. It defines the place of the metering process in the network path.

Observation Point: The observation can be made anywhere along the media path or on the endpoints themselves. The observation is only relevant in a unidirectional sense.

Element Data Type: unsigned8

Element Semantics: identifier

Element Units: n/a

Element Range Begin: 0

Element Range End: 0xFF

Element Id: TBDperfObservationType

Status: current

Use and Applications

0: unknown: The Metering Process does not specify the observation type

1: receiver: The Metering Process is, or is co-located with, the receiver of the RTP stream.

2: sender: The Metering Process is, or is co-located with, the sender of the RTP stream.

3: passive: The Metering Process passively observed the RTP stream.

4: rtcp: The Metering Process obtains the data conveyed in the IPFIX message for one or more RTCP reports.

Calculation Method:

Units of Measurement: n/a

Measurement Timing

4.1.2. perfIntervalStartMilliseconds

Name: perfIntervalStartMilliseconds

Description: Start time of the monitoring interval in milliseconds since 0000 UTC Jan 1, 1970. The time is taken from the local clock which SHOULD be NTP synchronized. If a flow only covers part of the monitoring interval (for example, the flow started after the interval start time), start time MUST be set to the start time of the monitoring interval.

Observation Point:

Element Data Type: dateTimeMilliseconds

Element Semantics: identifier

Element Units: n/a

Element Range Begin: 0

Element Range End: ???

Element Id: TBDperfIntervalStartMilliseconds

Status: current

Use and Applications

Calculation Method:

Units of Measurement: n/a

Measurement Timing

4.1.3. perfIntervalEndMilliseconds

Name: perfIntervalEndMilliseconds

Description: End time of the flow's monitoring interval in milliseconds since 0000 UTC Jan 1, 1970. The time is taken from the local clock which SHOULD be NTP synchronized. If the flow covers part of the monitoring interval (for example, the flow ended before the interval end time), the perfIntervalEndMilliseconds MUST be set to the end of observation interval.

Observation Point:

Element Data Type: datetimeMilliseconds

Element Semantics: identifier

Element Units: n/a

Element Range Begin: 0

Element Range End: 0xFF

Element Id: TBDperfObservationType

Status: current

Use and Applications

0: unknown: The Metering Process does not specify the observation type

1: receiver: The Metering Process is, or is co-located with, the receiver of the RTP stream.

2: sender: The Metering Process is, or is co-located with, the sender of the RTP stream.

3: passive: The Metering Process passively observed the RTP stream.

4: rtcp: TBD

Calculation Method:

Units of Measurement: n/a

Measurement Timing

4.1.4. perfSampleOffsetMilliseconds

Name: perfSampleOffsetMilliseconds

Description: Offset of the observation interval contained in this flow record. The value is measured in milliseconds and contains the offset of the beginning of the flow record from the beginning of the RTP stream.

Observation Point:

Element Data Type: unsigned32

Element Semantics: identifier

Element Units: milliseconds

Element Range Begin: 0

Element Range End: 0xFFFFFFFF

Element Id: TBDperfSampleOffsetMilliseconds

Status: current

Use and Applications

Calculation Method:

Measurement Timing

4.1.5. perfSampleTimeMilliseconds

Name: perfSampleTimeMilliseconds

Description: An IPFIX observer may generate and export a flow record for the entire duration of an RTP stream or for a specific part, e.g. a fixed time slice of 10 seconds. In case a single flow record is created the rtpSampleTime equals the RTP stream duration in milliseconds. In either case the rtpStreamState IE should be set to true if this flow record describes an ended RTP stream.

Observation Point:

Element Data Type: unsigned32

Element Semantics: DeltaCounter

Element Units: milliseconds

Element Range Begin: 0

Element Range End: 0xFFFFFFFF

Element Id: TBDperfSampleTimeMilliseconds

Status: current

Use and Applications

Calculation Method:

Units of Measurement: milliseconds

Measurement Timing

4.1.6. perfStreamState

Name: perfStreamState

Description: Using the rtpSampleOffset and rtpSampleTime IEs flow entries may be generated which describe only part of an RTP stream. This IE is used to describe the state of the observed stream, e.g. to indicate the reception of the last flow record belonging to a single RTP stream.

Observation Point:

Element Data Type: unsigned8

Element Semantics: identifier

Element Units: n/a

Element Range Begin: 0

Element Range End: 0xFF

Element Id: TBDperfObservationType

Status: current

Use and Applications

0: undefined: The state of the stream is not known.

1: running: The Metering Process expects more RTP packets or has already received packets for this RTP stream which are outside the scope of this flow record.

2: ended: The Metering Process determined that the RTP stream ended. Information sources could be signaling information or the fact that no RTP media has been received for a longer period of time.

3: no packets: The Metering Process has not received any RTP packets for this RTP stream in the observation interval but the stream has not ended. A VoIP endpoint may have requested the media stream to be suspended, i.e. put ’on hold’ (tbd:reference to sendonly ..)

Calculation Method:

Units of Measurement: n/a

Measurement Timing

4.1.7. perfPacketLoss

Name: perfPacketLoss

Description: The packet loss metric reports the number of individual packets that were lost in the reporting interval.

Observation Point: The observation can be made anywhere along the media path or on the endpoints them selves. The observation is only relevant in a unidirectional sense.

Element Data Type: unsigned32

Element Semantics: deltaCounter

Element Units: packets

Element Range Begin: 0

Element Range End: 0xFFFFFFFE

Element Id: TBDperfPacketLoss

Status: current

4.1.8. perfPacketExpected

Name: perfPacketExpected

Description: The number of packets there were expected within a monitoring interval.

Observation Point: The observation can be made anywhere along the media path or on the endpoints them selves. The observation is only relevant in a unidirectional sense.

Element Data Type: unsigned32

Element Semantics: deltaCounter

Element Units: packets

Element Range Begin: 0

Element Range End: 0xFFFFFFFE

Element Id: TBDperfPacketExpected

Status: current

4.1.9. perfPacketLossRate

Name: perfPacketLossRate

Description: Percentage of number of packets lost out of the total set of packets sent.

Observation Point: The observation can be made anywhere along the media path or on the endpoints them selves. The observation is only relevant in a unidirectional sense.

Element Data Type: unsigned16

Element Semantics: quantity

Element Units: float16 (IPFIX has not defined float16 yet)

Element Range Begin: 0

Element Range End: 0x64

Element Id: TBDperfPacketLossRate

Status: current

4.1.10. perfPacketLossEvent

Name: perfPacketLossEvent

Description: The packet loss event metric reports the number of continuous sets of packets that were lost in the reporting interval.

Observation Point: The observation can be made anywhere along the media path or on the endpoints them selves. The observation is only relevant in a unidirectional sense.

Element Data Type: unsigned32

Element Semantics: deltaCounter

Element Units: event

Element Range Begin: 0

Element Range End: 0xFFFFFFFE

Element Id: TBDperfPacketExpected

Status: current

4.1.11. perfPacketInterArrivalJitterAvg

Name: perfPacketInterArrivalJitterAvg

Description: This metric measures the absolute deviation of the difference in packet spacing at the measurement point compared to the packet spacing at the sender.

Observation Point: The observation can be made anywhere along the media path or on the receiver. The observation is only relevant in a unidirectional sense.

Element Data Type: unsigned32

Element Semantics: quantity

Element Units: microseconds

Element Range Begin: 0

Element Range End: 0xFFFFFFFE

Element Id: TBDperfPacketInterArrivalJitterAvg

Status: current

4.1.12. perfPacketInterArrivalJitterMin

Name: perfPacketInterArrivalJitterMin

Description: This metric measures the minimum value the calculation used for perfPacketInterArrivalJitterAvg within the monitoring interval.

Observation Point: The observation can be made anywhere along the media path or on the receiver. The observation is only relevant in a unidirectional sense.

Element Data Type: unsigned32

Element Semantics: quantity

Element Units: microseconds

Element Range Begin: 0

Element Range End: 0xFFFFFFFE

Element Id: TBDperfPacketInterArrivalJitterMin

Status: current

4.1.13. perfPacketInterArrivalJitterMax

Name: perfPacketInterArrivalJitterMax

Description: This metric measures the maximum value the calculation used for perfPacketInterArrivalJitterAvg within the monitoring interval.

Observation Point: The observation can be made anywhere along the media path or on the receiver. The observation is only relevant in a unidirectional sense.

Element Data Type: unsigned32

Element Semantics: quantity

Element Units: microseconds

Element Range Begin: 0

Element Range End: 0xFFFFFFFE

Element Id: TBDperfPacketInterArrivalJitterMax

Status: current

4.1.14. rtpPacketizationTime

Name: rtpPacketizationTime

Description: The RTP audio packetization time defines the amount of audio contained in the individual RTP packet. This packetization is typically fixed for the duration of an RTP stream but may be changed. The allowed values depend on the codec. Values typically observed are 10, 20 or 30ms.

Depending on the codec the amount of data contained in each RTP packet can be derived from RTP time stamp information or RTP payload size.

If the packetization time changes during an IPFIX monitoring interval this value should indicate the most common value monitored. Optionally the rtpPacketizationChange Information Element may be updated accordingly.

Observation Point: The observation can be made anywhere along the media path or on the receiver. The observation is only relevant in a unidirectional sense.

Element Data Type: unsigned8

Element Semantics: quantity

Element Units: milliseconds

Element Range Begin: 0

Element Range End: 0xFF

Element Id: TBDrtpPacketizationTime

Status: current

4.1.15. rtpPacketizationChange

Name: rtpPacketizationChange

Description: Each time the packetization time of the observed RTP stream changes during the monitoring interval this IE is incremented.

Observation Point: The observation can be made anywhere along the media path or on the receiver. The observation is only relevant in a unidirectional sense.

Element Data Type: unsigned32

Element Semantics: deltaCounter

Element Units: n/a

Element Range Begin: 0

Element Range End: 0xFFFFFFFF

Element Id: TBDrtpPacketizationChange

Status: current

4.1.16. perfDuplicates

Name: perfDuplicates

Description: Packets belonging to an observed stream or session may be duplicated. The reason or source of duplication (e.g. the generator or entities on the network path) is out of scope of this IE. This IE describes the number of protocol specific duplicate packets observed in the monitoring interval.

Observation Point: anywhere

Element Data Type: unsigned32

Element Semantics: deltaCounter

Element Units: packets

Element Range Begin: 0

Element Range End: 0xFFFFFFFE

Element Id: TBDperfDuplicates

Status: current

Use and Applications

Calculation Method: The calculation method for duplicate packets depends on the transport and application protocol used. Duplicates on the application layer SHALL be counted if possible.

For [RFC3550] style RTP streams the RTP sequence numbers MUST be used to identify duplicate packets. If a packet with the same sequence number is observed twice or more in the monitoring interval it is counted as duplicate. The perfDuplicates IE describes the number of duplicate packets received, not counting the first packet with each sequence number.

Units of Measurement: packets

Measurement Timing n/a

4.1.17. rtpPacketOrder

4.1.18. rtpSequenceError

4.1.19. perfRoundTripNetworkDelay

Name: perfRoundTripNetworkDelay

Description: This metric measures the network round trip time between end stations for a flow.

Observation Point: The observation can be made anywhere along the flow path as long as the bidirectional network delay is accounted for.

Element Data Type: unsigned32

Element Semantics: quantity

Element Units: microseconds

Element Range Begin: 0

Element Range End: 0xFFFFFFFE

Element Id: TBDperfRoundTripNetworkDelay

Status: current

4.2. User and Application Layer

4.2.1. perfSessionSetupDelay

Name: perfSessionSetupDelay

Description: The Session Setup Delay metric reports the time taken from a request being initiated by a host/endpoint to the response (or request indicator) to the request being observed. This metric is defined in [RFC4710], however the units have been updated to microseconds.

Observation Point: This metric needs to be calculated where both request and response can be observed. This could be at network choke points, application proxies, or within the end systems themselves.

Element Data Type: unsigned32

Element Semantics: quantity

Element Units: microseconds

Element Range Begin: 0

Element Range End: 0xFFFFFFFE

Element Id: TBDperfSessionSetupDelay

Status: current

4.3. RTP Header

4.3.1. rtpProtocolVersion

Name: rtpProtocolVersion

Description: Value of the RTP version taken from the RTP header. For [RFC3550] RTP packets this will typically be set to 2.

Observation Point: anywhere

Element Data Type: unsigned8

Element Semantics: identifier

Element Units: none

Element Range Begin: 0

Element Range End: 0x02

Element Id: TBDrtpProtocolVersion

Status: current

Use and Applications The RTP protocol version is taken directly from the RTP header information. It can be used to identify RTP packets and differ between different RTP versions once they become available.

Calculation Method: The value is obtained from the RTP header. For [RFC3550] RTP this two bit field must always be set to two (2). In case different values are observed in a single monitoring interval the IE shall carry the value identified in the first RTP packet of the monitoring interval.

Units of Measurement: none

Measurement Timing does not apply.

4.3.2. rtpSSRC

Name: rtpSSRC

Description: Value of the synchronization source (SSRC) field in the RTP header of the flow. This field is defined in [RFC3550].

Observation Point: This metric can be gleaned from the RTP packets directly, so the observation point can be either on the any RTP endpoints or on the flow path in between the endpoints. It is possible for the SSRC to change for a media flow without notice. In these cases the IE would represent the value seen in the packet-- the new SSRC and this would be treated as a new 'flow' per configured flow record definitions.

Element Data Type: unsigned32

Element Semantics: identifier

Element Units: none

Element Range Begin: 0

Element Range End: 0xFFFFFFFE

Element Id: TBDrtpSSRC

Status: current

Use and Applications The RTP SSRC value denotes a specific media stream. As such when trying to differentiate media stream problems between session participants the SSRC field is needed.

Calculation Method: Copy from RTP header's SSRC field as defined in [RFC3550]. In the case of a non-RTP flow, or the time period in which the flow has not been verified to be a RTP flow the value 0xFFFFFFFE MUST be reported.

Units of Measurement: identifier

Measurement Timing It is possible that the SSRC may have be renegotiated mid-session due to collisions with other RTP senders.

4.3.3. rtpPayloadType

Name: rtpPayloadType

Description: The value of the RTP Payload Type Field as observed in the RTP header of the flow. This field is defined in [RFC3550]

Observation Point: This metric can be gleaned from the RTP packets directly, so the observation point needs to on the flow path or within the endpoints.

Element Data Type: unsigned8

Element Semantics: identifier

Element Units: none

Element Range Begin: 0

Element Range End: 0xFF

Element Id: TBDrtpPayloadType

Status: current

4.3.4. rtpMediaType

Name: rtpMediaType

Description: The rtpMediaType field carries the verbatim media type name (e.g. Audio) as defined by [RFC4855].

Observation Point: anywhere

Element Data Type: string

Element Semantics: tbd

Element Units: n/a

Element Range Begin: n/a

Element Range End: n/a

Element Id: TBDrtpMediaType

Status: current

4.3.5. rtpMediaSubType

Name: rtpMediaSubType

Description: The rtpMediaSubType field carries the verbatim media type name (e.g. PCMA) as defined by [RFC4855].

Observation Point: anywhere

Element Data Type: string

Element Semantics: tbd

Element Units: n/a

Element Range Begin: n/a

Element Range End: n/a

Element Id: TBDrtpMediaSubType

Status: current

4.3.6. RTP Payload

This section defines additional Information Elements which describe RTP stream payload and characteristics beyond the transport information. Complicated metrics may be subject to different measurement methods. In order to prevent data from being unusable due to incompatible formats or measurement methods most Information Elements are counter values which allow calculation of metrics on mediator or collector systems. Additionally this allows matching flow records to be aggregated by addition, e.g. addition of the rtpPacketCount values of multiple observation intervals.

4.3.6.1. rtpPacketCount

Name: rtpPacketCount

Description: Number of RTP packets covered in this flow record. This includes observed duplicate packets.

Observation Point: anywhere

Element Data Type: unsigned32

Element Semantics: deltaCounter

Element Units: packets

Element Range Begin: 0

Element Range End: 0xFFFFFFFE

Element Id: TBDrtpPacketCount

Status: current

Use and Applications The packet count may be used in conjunction with the rtpPacketCountLoss and rtpPacketCountDiscard information elements to calculate a packet loss rate per monitoring interval. The benefit of transporting absolute numbers versus percentiles is that an IPFIX mediator or collector may merge multiple IPFIX records of the same or different RTP streams into a single record for aggregation purposes.

Calculation Method: The IPFIX observer counts all packets belonging to the respective flow. Lost packets as identified by skipped RTP sequence numbers MUST not be counted. Duplicate packets MUST be counted. The packet order is not observed and does not impact the packet count.

Units of Measurement: packets

Measurement Timing

4.3.6.2. rtpPacketCountLoss

Name: rtpPacketCountLoss

Description: Number of RTP packets lost in the duration covered by this flow record. The number of lost packets SHOULD be calculated using the RTP sequence numbers.

Observation Point: anywhere

Element Data Type: unsigned32

Element Semantics: deltaCounter

Element Units: packets

Element Range Begin: 0

Element Range End: 0xFFFFFFFE

Element Id: TBDrtpPacketCountLoss

Status: current

Use and Applications

Calculation Method: The IPFIX observer tracks the RTP sequence numbers of each RTP stream and at the end of the monitoring interval counts the number of packets not received based on the missing sequence numbers.

Units of Measurement: packets

Measurement Timing

4.3.6.3. rtpPacketCountDiscard

Name: rtpPacketCountDiscard

Description: Passive monitoring equipment shall assume a fixed 40 millisecond jitter buffer (TODO: add reference to TM Forum/ITU). A packet observed later than the expected packet inter-arrival time plus the 40ms is assumed to be received by the RTP receiver too late to be played out. Even though the packet may be received by the RTP receiver it will be discarded which has the same effect as packet loss.

Observation Point: anywhere

Element Data Type: unsigned32

Element Semantics: deltaCounter

Element Units: packets

Element Range Begin: 0

Element Range End: 0xFFFFFFFE

Element Id: TBDrtpPacketCountDiscard

Status: current

Use and Applications

Calculation Method: The IPFIX observer implements a 40ms jitter buffer per RTP stream observing sequence numbers as an RTP endpoint would do. Packets received 40ms after their scheduled playout time are considered discarded. Lost packets MUST not be counted as discarded.

Units of Measurement: packets

Measurement Timing

4.3.7. rtpMediaType

4.3.8. rtpMediaSubType

4.3.9. rtpDelayType

4.3.10. rtpDelayOneWay

4.3.11. rtpIsSRTP

4.3.12. rtpTimestamp

4.3.13. rtpCodecChange

4.3.14. rtpMarkerBit

4.3.15. rtpComfortNoise

4.3.16. rtpDSCPChange

5. Security Considerations

The recommendations in this document do not introduce any additional security issues to those already mentioned in [RFC5101] and [RFC5477]

6. IANA Considerations

This document requires an elements assignment to be made by IANA.

7. Acknowledgements

The authors would like to thank Shingo Kashima, Jan Novak and Al Morton for their invaluable review and comments. Thank-you.

8. References

8.1. Normative References

[RFC5101] Claise, B., "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information", RFC 5101, January 2008.

[RFC5610] Boschi, E., Trammell, B., Mark, L., and T. Zseby, "Exporting Type Information for IP Flow Information Export (IPFIX) Information Elements", RFC 5610, July 2009.

[RFC4710] Siddiqui, A., Romascanu, D., and E. Golovinsky, "Real-time Application Quality-of-Service Monitoring (RAQMON) Framework", RFC 4710, October 2006.

[RFC5102] Quittek, J., Bryant, S., Claise, B., Aitken, P., and J. Meyer, "Information Model for IP Flow Information Export",

RFC 5102, January 2008.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July 2003.

[RFC3497] Gharai, L., Perkins, C., Goncher, G., and A. Mankin, "RTP Payload Format for Society of Motion Picture and Television Engineers (SMPTE) 292M Video", RFC 3497, March 2003.

[RFC5389] Rosenberg, J., Mahy, R., Matthews, P., and D. Wing, "Session Traversal Utilities for NAT (STUN)", RFC 5389, October 2008.

[RFC4855] Casner, S., "Media Type Registration of RTP Payload Formats", RFC 4855, February 2007.

[RFC6076] Malas, D. and A. Morton, "Basic Telephony SIP End-to-End Performance Metrics", RFC 6076, January 2011.

[iana-ipfix-assignments]
Internet Assigned Numbers Authority, "IP Flow Information Export Information Elements (http://www.iana.org/assignments/ipfix/ipfix.xml)".

8.2. Informative References

[I-D.trammell-ipfix-ie-doctors]
Trammell, B. and B. Claise, "Guidelines for Authors and Reviewers of IPFIX Information Elements", draft-trammell-ipfix-ie-doctors-03 (work in progress), October 2011.

[RFC2508] Casner, S. and V. Jacobson, "Compressing IP/UDP/RTP Headers for Low-Speed Serial Links", RFC 2508, February 1999.

[RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, March 2004.

[RFC2250] Hoffman, D., Fernando, G., Goyal, V., and M. Civanlar, "RTP Payload Format for MPEG1/MPEG2 Video", RFC 2250, January 1998.

[RFC2890] Dommety, G., "Key and Sequence Number Extensions to GRE", RFC 2890, September 2000.

[RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, December 2005.

[RFC5761] Perkins, C. and M. Westerlund, "Multiplexing RTP Data and Control Packets on a Single Port", RFC 5761, April 2010.

[I-D.huici-ipfix-sipfix] Huici, F., Niccolini, S., and S. Anderson, "SIPFIX: Use Cases and Problem Statement for VoIP Monitoring and Exporting", draft-huici-ipfix-sipfix-00 (work in progress), June 2009.

[RFC2321] Bressen, A., "RITA -- The Reliable Internetwork Troubleshooting Agent", RFC 2321, April 1998.

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

[RFC5477] Dietz, T., Claise, B., Aitken, P., Dressler, F., and G. Carle, "Information Model for Packet Sampling Exports", RFC 5477, March 2009.

[VoIP-monitor] L. Chang-Yong, H. Kim, K. Ko, J. Jim, and H. Jeong, "A VoIP Traffic Monitoring System based on NetFlow v9, International Journal of Advanced Science and Technology, vol. 4, Mar. 2009".

Authors' Addresses

Aamer Akhter
Cisco Systems, Inc.
7025 Kit Creek Road
RTP, NC 27709
USA

Email: aakhter@cisco.com

Hendrik Scholz
VOIPFUTURE GmbH
Wendenstrasse 4
Hamburg 20097
Germany

Phone: +49 40 688 900 100
Email: hscholz@voipfuture.com
URI: http://www.voipfuture.com/

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 28, 2012

A. Akhter
Cisco Systems
March 27, 2012

Methodology for Network Flow Performance Measurement
draft-akhter-opsawg-perfmon-method-02.txt

Abstract

There is a need to be able to quantify and report the performance of network applications and the network service in handling user data. This performance data provides information essential in validating service level agreements, fault isolation as well as early warnings of network greater problems. This document describes a generic methodology for calculating metrics related to network based applications. In addition, to the performance metrics, several additional information elements are included to help provide greater context to the reports. The measurements use audio/video applications as base examples but are not restricted to these class of applications.

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 http://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 September 28, 2012.

Copyright Notice

Table of Contents

1. Introduction

Today's networks support a multitude of highly demanding and sensitive network applications. Network issues are readily apparent by the users of these applications due to the sensitivity of these applications to impaired network conditions. Examples of these network applications include applications making use of IP based audio, video, database transactions, virtual desktop interface (VDI), online gaming, cloud services and many more. In some cases, the impaired application translates directly to loss of revenue. In other cases, there may be regulatory or contractual service level agreements that motivate the network operator. Due to the sensitivity of these types of applications to impaired service, it leaves a poor impression of the network service on the user--regardless of the actual performance of the network itself. In the case of an actual problem within the network service, monitoring the performance may yield an early indicator of a much more serious problem.

Due to the demanding and sensitive nature of these applications, network operators have tried to engineer their networks towards wringing better and differentiated performance. However, that same differentiated design prevents network operators from extrapolating observational data from one application to another, or from one set of synthetic (active test) test traffic to actual application performance. This gap highlights the importance of generic measurements as well as the reliance on user traffic measurents--rather than synthetic tests.

Performance measurements on user data provide greater visibility not only into the quality of experience of the end users but also visibility into network health. With regards to network health, as flow performance is being measured, there will be visibility into the end to end performance which means that not only visibility into local network health, but also viability into remote network health. If these measurements are made at multiple points within the network (or between the network and end device) then there is not only identification that there might be an issue, but a span of area can be established where the issue might be. The resolution of the fault increases with the number of measurement points along the flow path.

The IP Flow Information Export Protocol (IPFIX) [RFC5101] provides new levels of flexibility in reporting from measurement points across the life cycle of a network based application. IPFIX can provide granular results in terms of flow specificity as well as time granularity. At the same time, IPFIX allows for summarization of data along different types of boundaries for operators that are unconcerned about specific sessions but about health of a service or

a portion of the network. This document details the methodlogy of measurement, while an accompanying document describes the expression of the measurements in IPFIX format.

Where possible, an attempt has been made to make use of existing definitions of metrics ([RFC4710]) and if needed, clarify and expand on them to widen their usage with additional applications, and network devices. For example, the RTP measurments have generally defined from the prespective of end systems rather than intermdiate nodes which are not alwyas privy to the application context and may have limited scaling properties. The methodology described in [I-D.ietf-pmol-sip-perf-metrics] is used to describe the methodology of measurement.

There has been related work in this area such as [RFC2321]. [I-D.huici-ipfix-sipfix], and [VoIP-monitor]. This document is also an attempt to generalize as well as standardize the reporting formats and measurement methodology.

2. Terminology

Terms used in this document that are defined in the Terminology section of the IPFIX Protocol [RFC5101] document are to be interpreted as defined there.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

In addition, the information element definitions use the following terms:

Name: Name of the information element

Description: Short description of what the information element is trying to convey.

Observation Point: Where the measurement is meant to be performed. Either at an intermediate system (for example, a router) or end system.

Use and Applications An explanation of how this particular information element would be used.

Calculation Method: In the case of metrics, this section describes how the metric is calculated, as well as any special conditions.

Units of Measurement: In the case of metrics, what are the units of measurement. The text here is expected to be wider and more descriptive than in the IPFIX Element Units section.

Measurement Timing: Discussion on the acceptable range of timing and sampling intervals.

3. General Usage

3.1. Quality of Service (QoS) Monitoring

The network operator needs to be able to gauge the end user's satisfaction with the network service. While there are many components of the satisfaction such as pricing, packaging, offering, etc., a major component of satisfaction is delivering a consistent service. The user builds trust on this consistency of the network service and runs network applications with confidence-- which is of course the end goal. Without the ability to deliver a consistent service for end user network applications network operator will be left dealing with price sensitive disgruntled users with very low expectations and utilization (if they don't have choice of operator) or abandonment (if they have choice).

3.2. Service Level Agreemnt (SLA) Validation

Similar to QoS and QoE validation, there might be contractual or regulatory requirements that need to be met by the network operator. Monitoring the performance of the flows allows the application operator, network operator as well as the end user to validate if the target service is being delivered. While there is quite a diversity in the codification of network SLAs, the eventually involve some measurement of network uptime, end to end latency, end to end jitter and perhaps service response time. In the case of SLA violation, the start and end times, nature and network scope of the violation needs to be captured to allow for the most accurate settling of the SLA violation.

3.3. Fault Isolation and Troubleshooting

It has been generally easier to troubleshoot and fix problems that are binary in nature: it either works or does not work. The host is pingable or not pingable. However, it is the much more difficult to resolve transitory issues that move from location to location, are not complete faiures and sometimes with unverfifiable end user

reports as the only indication of a problem. It is these intermittent and seemingly inconsistent network impairments that performance metrics can be extremely helpful with. Just the basic timely detection that there is a problem (or an impending problem) can give the operator provider the confidence that there is a real problem that needs to be resolved. The next step would be to assist the operator in a speedy resolution by providing information regarding the network location and nature of the problem.

4. New Information Elements

The information elements are organized into two main groups:

Transport Layer: Metrics that might be calculated from observations at higher layers but essentially provide information about the network transport of user date. For example, the metrics related to packet loss, latency and jitter would be defined here.

User and Application Layer: Metrics that are might be affected by the network indirectly, but are ultimately related to user, end-system and session states. For example, session setup time, transaction rate and session duration would be defined here.

Contextual Elements: Information elements that provide further context to the metrics. For example, media type, codec type, and type of application would be defined here.

4.1. Transport Layer

4.1.1. perfPacketLoss

Name: perfPacketLoss

Description: The packet loss metric reports the number of individual packets that were lost in the reporting interval.

Observation Point: The observation can be made anywhere along the media path or on the endpoints them selves. The observation is only relevant in a unidirectional sense.

Use and Applications The packet loss metric can be used to determine if there is a network impairment that is causing packet loss upstream of the measurement point. When there are observation points on either side of the impairment location it is possible to locate the impairment. With the location information the operator can is able to perform quicker fault-isolation as well as shorten time to resolution. Depending on implementation and operator

configuration, the granularity of contextual information can be very specific. For example, these traffic loss statistics when sent with IP subnet or DSCP information can provide visibilty into QoS specific or network topology issues.

Calculation Method: This metric requires that each IP packet be individually marked with a monotonically incrementing sequence number. A number of encapsulations support this type of sequencing: IPSec ESP [RFC4303], GRE [RFC2890] and RTP [RFC3550]. An analysis of the sequence number field can yield the lost number of packets. In certain cases, there might be an element of discovery and synchronization of the flow itself before the measurement can be made. An example of this can be found for RTP flows running on ephemeral UDP port numbers. In these cases, reporting 0 as packet loss would be misleading and the value 0xFFFFFFFF MUST be used in cases where the packet loss value cannot be determined. In the case of a monitor interval where synchronization was achieved mid-interval, the loss packet counter MAY be used to represent the remainder of the interval. As this metric is a deltaCounter, the number of loss packets only represent the observation within the reporting interval. Due to the dependency on the arrival of a packet with a sequence number to calculate loss, the loss calculation may be indefinitely delayed if no more packets arrive at all. For the case of RTP, in addition to the 16 bit sequence number field in RFC3550, there is also the additional 16-bit high-order sequence number field (for a total of 32-bit seq number space) that is used in RFC3497 [RFC3497]. RFC3497 traffic runs at a very high rate and the 32-bit field allow for additional time for wrapping (21 seconds). So, a loss span of greater than 21 seconds measured only by the 16-bit field will lead to inaccurate reporting. In the case of secure RTP [RFC3711], the relevant portion of the RTP header is in the clear and lost packet counting can still be performed. It is important to note that the sequence number space is unique per RTP SSRC. Therefore it is important to track the high sequence number seen on a per SSRC-5-tuple basis. There may be multiple SSRCs in a single 5-tuple. Certain applications inject non-RTP traffic into the same 5-tuple as the media stream. RTCP packets may be seen in the same 5-tuple as the RTP stream [RFC5761], and STUN [RFC5389] packets may also be seen. The loss detection should ignore these packets. There may be spans within the network where header compression schemes such as [RFC2508] are used. In cases where the measurement device is terminating the compression, and the measurement implementation does not support calculation of the metric the value 0xFFFFFFFF MUST be reported. In other cases the measurement point may be at a midpoint of the header compression network span. Depending on the mechanics of header compression, sequencing information may be present and it is possible to

calculate the metric. In such cases the implementation SHOULD perform the calculation and report the metric.

Units of Measurement: packets

Measurment Timing To be able to calculate this metric a continuous set of the flow's packets (as each would have an incrementing sequence number) needs to be monitored. Therefore, per-packet sampling would prevent this metric from being calculated. However, there are other sampling methodologies that might be usable. It is possible to generate sampled metrics by sampling spans of continuous packets, however a portion of the span may have to be utilized for resynchronization of the sequence number. Another form of acceptable sampling would be at the flow level.

4.1.2. perfPacketExpected

Name: perfPacketExpected

Description: The number of packets there were expected within a monitoring interval.

Observation Point: The observation can be made anywhere along the media path or on the endpoints them selves. The observation is only relevant in a unidirectional sense.

Use and Applications The perfPacketExpected is a mid-calculation metric used in the generation of perfPacketLossRate. It is equivilent to the highest received packet sequence number at the time of measurement. As the value only increments when packets are received, packet loss may be occouring at the time of measurement but perfPacketExpected remains constant.

Calculation Method: The subtraction of the last sequence number from the first sequence number in monitoring interval yields the expected count. As discussed with perfPacketLost, there might be a delay due to synchronization with the flow's sequence numbers and in such times the value of the metric should be set to 0xFFFFFFFE. Care has to be taken to account for cases where the packet's sequence number field wraps. For RTP, the expected count calculation formula can be found in Appendix A.3 of [RFC3550]. Refer to the perfPacketLoss metric regarding considerations for header compression. The value 0xFFFF is used to represent cases where the metric could not be calculated.

Units of Measurement: packets

Measurment Timing Same considerations as perfPacketLoss

4.1.3. perfPacketLossRate

Name: perfPacketLossPercentage

Description: Percentage of number of packets lost out of the total set of packets sent.

Observation Point: The observation can be made anywhere along the media path or on the endpoints them selves. The observation is only relevant in a unidirectional sense.

Use and Applications The perfPacketLossRate metric can be used to normalize the perfPacketLoss metric to handle cases where different flows are running at different packet per second (PPS) rates. Due to the normalization, comparisons can now be made against thresholds (for creating alerts, etc.). In addition, the percentage form of the metric allows for comparisons against other flows at the same observation point to determine if there is an equal bias for drops between the flows. Otherwise, the perfoPacketLossRate is used in same way as perfPacketLoss. This value can be derived from perfPacketExpected and perfPacketLoss and is offered as a convenience to ease functions such as thresholding, and pre-computed reporting. It shoudl be noted that for large values of perfPacketExpected and perfPacketLoss it might be preferable and more accurate for the conversion to percentage to occour at a later stage where the accuracy can be controlled.

Calculation Method: The number of lost packets divided by the number of expected packets in an interval period multiplied by 100. In cases where perfPacketLoss is unknown (for example due to synchronization issues), the perfPacketLossRate would also be unknown. If there are multiple flows whose loss rate is being aggregated, then the average of the individual flows is used. Refer to the perfPacketLoss metric regarding considerations for header compression.

Units of Measurement: percentage

Measurment Timing Same notes as perfPacketLossPercentage

4.1.4. perfPacketLossEvent

Name: perfPacketLossEvent

Description: The packet loss event metric reports the number of continuous sets of packets that were lost in the reporting interval.

Observation Point: The observation can be made anywhere along the media path or on the endpoints them selves. The observation is only relevant in a unidirectional sense.

Use and Applications The perfPacketLossEvent metric can provide loss information for protocols that do not implement per packet sequencing. Similarly to the perfPacketLoss metric, the packet loss event metric can be used to determine if there is a network impairment that is causing packet loss upstream of the measurement point. In cases where both the perfPacketLoss and perfPacketLossEvent metric are available, the ratio between the packet loss and packet event count can provide the average loss length. The average loss length provides additional information regarding the cause of the loss. For example, a dirty fiber connection might have a low average loss length, while a routing protocol convergence will have a high loss length.

Calculation Method: This data value is a simplified version of the Lost Packets metric. Whereas Lost Packets counts individual packet loss, the 'loss event count' metric counts sets of packets that are lost. For example, in the case of a sequence of packets: 1,3,6,7,10 the packets marked 2,4,5,8 and 9 are lost. So, a total of 5 packets are lost. This same sequence translates to 3 loss events: (2), (4,5) and (8,9). In the case of RTP, the sequence number in the RTP header can be used to identify loss events. Certain protocols such as TCP and UDP+MPEG2-TS encapsulation in IP have sequencing information, but the sequence field is incremented by individual IP packets. As a side note, in the case of UDP+ MPEG2-TS encapsulation the simple use of RTP+MPEG2-TS via [RFC2250] results in the avaliability of the more granular perfPacketLoss metrics. In these cases, the perfPacketLoss metric cannot be calculated but the perfPacketLossEvent can be calculated and can provide detection of loss. The value 0xFFFFFFFF is used to represent non-applicable cases such as lack of sequence number synchronization. Many of the same considerations as for perfPacketLoss apply to perfPacketLoss event. Please refer to the Calculation Method section of the perfPacketLoss.

Units of Measurement: event counts

Measurment Timing Please refer to the measurement timing section of perfPacketLoss.

4.1.5. perfPacketInterArrivalJitterAvg

Name: perfPacketInterArrivalJitterAvg

Description: This metric measures the absolute deviation of the difference in packet spacing at the measurement point compared to the packet spacing at the sender.

Observation Point: The observation can be made anywhere along the media path or on the receiver. The observation is only relevant in a unidirectional sense.

Use and Applications The inter arrival jitter data value can be used be network operator to determine the network's impact to the spacing in between a media stream's packets as they traverse the network. For example, in the case of media applications, the receiving end system is expecting these packets to come in at a particular periodicity and large deviations may result in de-jitter buffers adding excessive delay, or the media packets being discarded. When the data is reported from multiple intermediate nodes, the area of the network that is having a detrimental contribution can be identified. On a non-media application level, the inter arrival jitter metrics can be used for early indication queuing contention within the network (which could lead to packet loss).

Calculation Method: The inter arrival jitter value makes use of the association of sending time with an IP packets and comparison of the arrival time on the monitoring point. In certain protocols, a representation of sending time is encoded into the header itself. For example, in the case of RTP packets, the RTP header's timestamps field represents encoder clock ticks-- which are representations of time. Similarly, in the case of TCP options encode absolute timestamps values. For RTP the calculation method can be found in Appendix A of [RFC3550]. It should be noted that the RFC3550 calculation is on the last 16 packets measured. The most recent value calculated SHOULD be reported at the end of the monitoring interval. The range of the jitter values during the monitoring interval can be reported using perfPacketInterArrivalJitterMin and perfPacketInterArrivalJitterMax. Similarly to the perfPacketLoss case there may be periods of time where the jitter value cannot be calculated. In these cases, the 0xFFFFFFFF value should be used

to convey the lack of availability of the metric. As mentioned earlier, the RTP header timestamps is actually a 'sample-stamp' (ie clicks) from the encoder's clock. The frequency of the clock is dependent on the codec. Some codecs (eg AAC-LD) support multiple possible frequencies one of which is then selected for the media-stream. The mapping to clock rate can be performed via mapping from the static RTP payload type (RTP-PT), but newer codecs are make use of the dynamic payload type range and the RTP-PT (in the dynamic case) cannot be used to determine the clock frequency. There are various methods by which the clock frequency (deep packet inspection of the signalling, manual configuration, etc.) can be associated to the calculation method. The frequency should be locked in the metering layer to a unique combination of the IP source, IP destination, IP protocol layer-4 ports, RTP-PT and SSRC. By strict RFC3550 definition, the SSRC is set to a specific encoder clock and it is the SSRC that should be tracked rather than payload type. However, in recent discussions it has been noted that there are RTP implementations that might change the encoder clock frequency while maintaining the SSRC value. An encoder frequency change will be accompanied by a different RTP-PT.

Units of Measurement: microseconds

Measurment Timing Please refer to the measurement timing section of perfPacketLoss.

4.1.6. perfPacketInterArrivalJitterMin

Name: perfPacketInterArrivalJitterMin

Description: This metric measures the minimum value the calculation used for perfPacketInterArrivalJitterAvg within the monitoring interval.

Observation Point: The observation can be made anywhere along the media path or on the receiver. The observation is only relevant in a unidirectional sense.

Use and Applications Please refer to the 'Use and Applications' section of perfPacketInterArrivalJitterAvg. This specific metric, along with perfPacketInterArrivalJitterMax, is to capture the range of measurements observed within a monitoring interval as the average function may hide extremes.

Calculation Method: Please see the perfPacketInterArrivalJitterAvg section for general calculation section. The average calculation is evaluated on a running basis over the last 16 packets and the entire monitoring interval is not covered. In this metric, the minimum value is taken over the entire monitoring interval.

Units of Measurement: microseconds

Measurment Timing Please refer to the measurement timing section of perfPacketLoss.

4.1.7. perfPacketInterArrivalJitterMax

Name: perfPacketInterArrivalJitterMax

Description: This metric measures the maximum value the calculation used for perfPacketInterArrivalJitterAvg within the monitoring interval.

Observation Point: The observation can be made anywhere along the media path or on the receiver. The observation is only relevant in a unidirectional sense.

Use and Applications Please refer to the 'Use and Applications' section of perfPacketInterArrivalJitterAvg. This specific metric, along with perfPacketInterArrivalJitterMin, is to capture the range of measurements observed within a monitoring interval as the average function may hide extremes.

Calculation Method: Please see the perfPacketInterArrivalJitterAvg section for general calculation section. The average calculation is evaluated on a running basis over the last 16 packets and the entire monitoring interval is not covered. In this metric, the maximum value is taken over the entire monitoring interval.

Units of Measurement: microseconds

Measurment Timing Please refer to the measurement timing section of perfPacketLoss.

4.1.8. perfRoundTripNetworkDelay

Name: perfRoundTripNetworkDelay

Description: This metric measures the network round trip time between end stations for a flow.

Observation Point: The observation can be made anywhere along the flow path as long as the bidirectional network delay is accounted for.

Use and Applications The perfRoundTripNetworkDelay metric can be used in multiple ways. If the applicaiton being monitored provides interactive feedback to the user the perfRoundTripNetworkDelay can be used to judge the 'liveliness' of the application experience. Other use cases may involve troubleshooting throughput issues where the transport protocol's throughtput is affected by network delay.

Calculation Method: perfRoundTripNetworkDelay can estimated by accounting for the network flight time between a transport protocol request and response. In the case of TCP, this would the time difference between the TCP SYN and ACK packets in the TCP handshake. It should be noted that at times other than the TCP handshake the time difference between TCP end station packet. For RTP (RFC3550) based applications, the network round trip can be calculated by analysis of hte RTCP sending and receive times.

Units of Measurement: microseconds

Measurment Timing Depending on the method used to calculate the round trip time, the measurment may only be possible at specific times during the session lifecycle. In time periods where the metric is not current 'not calculated' SHOULD be reported.

4.2. User and Application Layer

4.2.1. perfSessionSetupDelay

Name: perfSessionSetupDelay

Description: The Session Setup Delay metric reports the time taken from a request being initiated by a host/endpoint to the response (or request indicator) to the request being observed. This metric is defined in [RFC4710], however the units have been updated to microseconds.

Observation Point: This metric needs to be calculated where both request and response can be observed. This could be at network choke points, application proxies, or within the end systems themselves.

Use and Applications The session setup delay metric can measure the end user initial wait experience as seen from the network transaction level. The value will not only include the network flight time, but also includes the server response time and may be used to alert the operator in cases where the overall service is overloaded and thus sluggish, or within normal operating values.

Calculation Method: Measure distance in time between the first bit of request and the first bit of the response. For the case of SIP, please see Section 4.3.1 of [I-D.ietf-pmol-sip-perf-metrics]

Units of Measurement: microseconds

Measurment Timing This measurement can be sampled on a session by session basis. It may be advisable to set sample targets on a per source range - to destination basis. Due to the nature of measurement intervals, there may be a period of time (and thus measurement reports) in which the perfSessionSetupDelay value has not been calculated. In these cases the value 0xFFFFFFFE MUST be used and can be interpreted to mean not applicable. For measurement intervals after perfSessionSetupDelay has been calculated and the existing calculated perfSessionSetupDelay value SHOULD be sent if reporting only on that single session. However, if multiple sessions are summarized in the report then the average for perfSessionSetupDelay values calculated in the most recent interval SHOULD be used. The intention with this behavior is to acknowledge that the value has not bee calculated, and when it has provide the freshest values available.

4.3. Contextual Elements

4.3.1. mediaRTPSSRC

Name: mediaRTPSSRC

Description: Value of the synchronization source (SSRC) field in the RTP header of the flow. This field is defined in [RFC3550]

Observation Point: This metric can be gleaned from the RTP packets directly, so the observation point needs to on the flow path or within the endpoints.

Use and Applications The RTP SSRC value denotes a specific media stream. As such when trying to differentiate media stream problems between session participants the SSRC field is needed.

Calculation Method: Copy from RTP header's SSRC field as defined in [RFC3550]. In the case of a non-RTP flow, or the time period in which the flow has not been verified to be a RTP flow the value 0xFFFFFFFE MUST be reported.

Units of Measurement: identifier

Measurment Timing It is possible that the SSRC may have be renegotiated mid-session due to collisions with other RTP senders.

4.3.2. mediaRTPPayloadType

Name: mediaRTPPayloadType

Description: The value of the RTP Payload Type Field as seen in the RTP header of the flow. This field is defined in [RFC3550]

Observation Point: This metric can be gleaned from the RTP packets directly, so the observation point needs to on the flow path or within the endpoints.

Use and Applications The RTP PT conveys the payload format and media encoding used in the RTP payload. For simple cases, where the RTP PT is from the statically defined range this can lead to an understanding of type of media codec used. With the knowledge of the codec being used the degree of media impairment (given loss values and jitter) can be estimated better. However, for more recent codecs, the RTP dynamic range is used. In these cases the RTP payload values are dynamically negotiated. In the case of a non-RTP flow, or the time period in which the flow has not been verified to be a RTP flow, the value 0xFFFF MUST be reported.

Calculation Method: Copy from RTP header's RTP-PT field as defined in [RFC3550]

Units of Measurement: identifier

Measurment Timing

4.3.3. mimeType

Name: mimeType

Description: The mime type describes the content of the flow.

Observation Point: The ideal location of this metric is on the application generators and consumers. However, given application signalling inspection or static configuration it is possible that intermediate nodes are able to generate mime type (eg. codec name) information.

Use and Applications The mime type value conveys information regarding the content of a flow. For example, in the case of Audio/Video applications the name of the codec used to encode the media in the flow. Simply reporting loss and jitter measurements are useful for detection of network problems. However, judging the degree of the impact on the audio/video experience needs additional information. The most basic information is the codec being used which when coupled with per-codec knowledge of sensitivity to the transport metrics a better idea of the experience can be gained.

Calculation Method: The valid values for the mime type are listed on the IANA mime type registry. For Audio/Video codecs, there is a specific media-types registry. Analysis of the RTP payload type may lead to the determination of the media codec. However, with the use of the RTP dynamic payload type range the media information is not encoded into the data packet. For these cases, intermediate nodes may need to perform inspection of the signalling (SIP, H.323, RTSP, etc.). In cases where the mediaCodec cannot be determined, the value 'unknown' MUST be used.

Units of Measurement: identifier

Measurment Timing

5. Security Considerations

The recommendations in this document do not introduce any additional security issues to those already mentioned in [RFC5101] and [RFC5477]

6. Acknowledgements

The authors would like to thank Rahul Patel, Jan Novak, Al Morton, Brad Fawcett, Doug Manley and Shingo Kashima for their invaluable review and comments. Thank-you.

7. References

7.1. Normative References

[RFC5101] Claise, B., "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information", RFC 5101, January 2008.

[RFC5610] Boschi, E., Trammell, B., Mark, L., and T. Zseby, "Exporting Type Information for IP Flow Information Export (IPFIX) Information Elements", RFC 5610, July 2009.

[RFC4710] Siddiqui, A., Romascanu, D., and E. Golovinsky, "Real-time Application Quality-of-Service Monitoring (RAQMON) Framework", RFC 4710, October 2006.

[RFC5102] Quittek, J., Bryant, S., Claise, B., Aitken, P., and J. Meyer, "Information Model for IP Flow Information Export", RFC 5102, January 2008.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July 2003.

[RFC3497] Gharai, L., Perkins, C., Goncher, G., and A. Mankin, "RTP Payload Format for Society of Motion Picture and Television Engineers (SMPTE) 292M Video", RFC 3497, March 2003.

[RFC5389] Rosenberg, J., Mahy, R., Matthews, P., and D. Wing, "Session Traversal Utilities for NAT (STUN)", RFC 5389, October 2008.

[I-D.ietf-pmol-sip-perf-metrics] Malas, D. and A. Morton, "Basic Telephony SIP End-to-End Performance Metrics", draft-ietf-pmol-sip-perf-metrics-07 (work in progress), September 2010.

[iana-ipfix-assignments] Internet Assigned Numbers Authority, "IP Flow Information Export Information Elements (http://www.iana.org/assignments/ipfix/ipfix.xml)".

7.2. Informative References

[I-D.ietf-pmol-metrics-framework] Clark, A. and B. Claise, "Guidelines for Considering New Performance Metric Development", draft-ietf-pmol-metrics-framework-12 (work in progress), July 2011.

[RFC2508] Casner, S. and V. Jacobson, "Compressing IP/UDP/RTP Headers for Low-Speed Serial Links", RFC 2508, February 1999.

[RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, March 2004.

[RFC2250] Hoffman, D., Fernando, G., Goyal, V., and M. Civanlar, "RTP Payload Format for MPEG1/MPEG2 Video", RFC 2250, January 1998.

[RFC2890] Dommety, G., "Key and Sequence Number Extensions to GRE", RFC 2890, September 2000.

[RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, December 2005.

[RFC5761] Perkins, C. and M. Westerlund, "Multiplexing RTP Data and Control Packets on a Single Port", RFC 5761, April 2010.

[I-D.huici-ipfix-sipfix] Huici, F., Niccolini, S., and S. Anderson, "SIPFIX: Use Cases and Problem Statement for VoIP Monitoring and Exporting", draft-huici-ipfix-sipfix-00 (work in progress), June 2009.

[nProbe] "nProbe - NetFlow/IPFIX Network Probe (http://www.ntop.org/nProbe.html)".

[RFC2321] Bressen, A., "RITA -- The Reliable Internetwork Troubleshooting Agent", RFC 2321, April 1998.

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

[RFC5477] Dietz, T., Claise, B., Aitken, P., Dressler, F., and G. Carle, "Information Model for Packet Sampling Exports", RFC 5477, March 2009.

[VoIP-monitor] L. Chang-Yong, H. Kim, K. Ko, J. Jim, and H. Jeong, "A VoIP Traffic Monitoring System based on NetFlow v9, International Journal of Advanced Science and Technology, vol. 4, Mar. 2009".

Author's Address

Aamer Akhter
Cisco Systems, Inc.
7025 Kit Creek Road
RTP, NC 27709
USA

Email: aakhter@cisco.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 8, 2011

Y. Gu
Huawei

March 7, 2011

Policies and dynamic data migration in DC draft-gu-dc-management-problem-statement-00

Abstract

Virtualization provides Data Center with feasibility and improves the utilization of limited physical resource, e.g. switches/routers, servers and links. Virtual machines (VM) are allowed to migrate to any place in the Data Center. A variety of policies (e.g. ACL, firewalls, load balancers, IPS and QoS) are deployed in Data Center to guarantee the SLA the provider signed with their clients. Dynamic information, such as TCP Connection Table, dynamic ACLs and cumulated data, is generated on network devices. In order to keep running services uninterrupted while VM migrating, relevant policies and dynamic information, also need to migrate with VM.

This document describes some examples of the policies and dynamic information that need to migrate with VM, the influence if they are not migrated with VM, the problems that need to consider when migrate polices and dynamic information. It also describes some existing network management protocols standardized by IETF and the advantages and disadvantages of them for operating policies and dynamic information migration respectively. The goal is to justify that it is necessary for IETF to make effort on policy and dynamic information migration for large virtualized Data Center.

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 http://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 September 8, 2011.

Table of Contents

1. Introduction

Data centers can host tens or even thousands of different applications. Some are simple applications such as web servers providing static content, while some may be very complex, e.g. e-commerce, that requiring all around privacy protection and data security. Clients of Data Center, unlike server hosting clients, raise more strict QoS and Security requirements. Clients may sign Service Level Agreement (SLA) with Data Center Provider to make sure their requirement can be guaranteed. To satisfy different level of security requirements and to manage and improve the performance of these applications, data centers typically deploy a large variety of middleboxes, including firewalls, load balancers, SSL offloaders, web caches, and intrusion prevention boxes.

To satisfy QoS requirements, Data Center also implement QoS mechanism as ISP network. For example, to deploy polices on Switches to execute traffic classification and marking. IEEE 802.1 DCB working group defines a series of standards to guarantee quality of service.

802.1Qau - Congestion Notification

802.1Qaz - Enhanced Transmission Selection

802.1Qbb - Priority-based Flow Control

Without regard to mobile network, the existing DC network management has a pre-assumption that the end hosts will not move. If an end host moves, because the physical link has to break down and the service also has to break down, the network can treat it as two separated parts: one host leave the network and another host join the network.

Server Virtualization and Virtual Machine Migration changes the situation and break the preassumption. Server Virtualization is not a new technology. But, because Cloud services become popular, which requires flexible resource assignment and effectively resource integration, server virtualization revitalizes again. Using server virtualization technologies, network adminitrator can reduce networking cost. To support the same volume of services, fewer network devices, servers and links are required than before. Multiple Virtual Machines (VMs) are established within a single physical server and the VMs are allowed to relocate to a different servers within the same subnet of Data Center, or even among different sites of a Data Center. This is so called VM Migration. VM Migration brings flexibility to Data Center, meanwhile it makes network management more complex and challenging.

While VM migrates, a very important requirement is that running services on the VM mustn't been interrupted. Though a 'zero delay' on running services is not realistic, but the services should be able to continue after a very short delay.

In order to avoid service interruption and minimize delay on running services, polices and dynamic information on network devices must be migrated timely and accurately. The policies and dynamic information includes those on switches, routers and middleboxes.

In the following section, we describe the policies and dynamic information migration on several example network devices. The influence to running services if they are not migrated accurately and timely. Then we will introduce the limitations of existing network management protocol.

2. Terminologies and concepts

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].

Source Network Device, Source switch, or Source device: the network device/switch/device from where the VM migrates. I.E. VM is originally located under the source network device/switch/device.

Destination Network Device, Destination switch, or Destination device: the network device/switch/device to where the VM migrates. I.E. VM is relocated to the destination network device/switch/device.

TCP connection table: A table containing TCP connection-specific information.

3. Policies on several network devices

In this draft, our discussion using the following figure as an example networking. The links between AGG1/AGG2 and Gateway2, AGG3/ AGG4 and Gateway1 are omitted for simplicity. the new VM1 under Server4 represents the VM1 after migration. VM1 and new VM1 don't exist at the same time. In the real world, the networking of DC could be different.

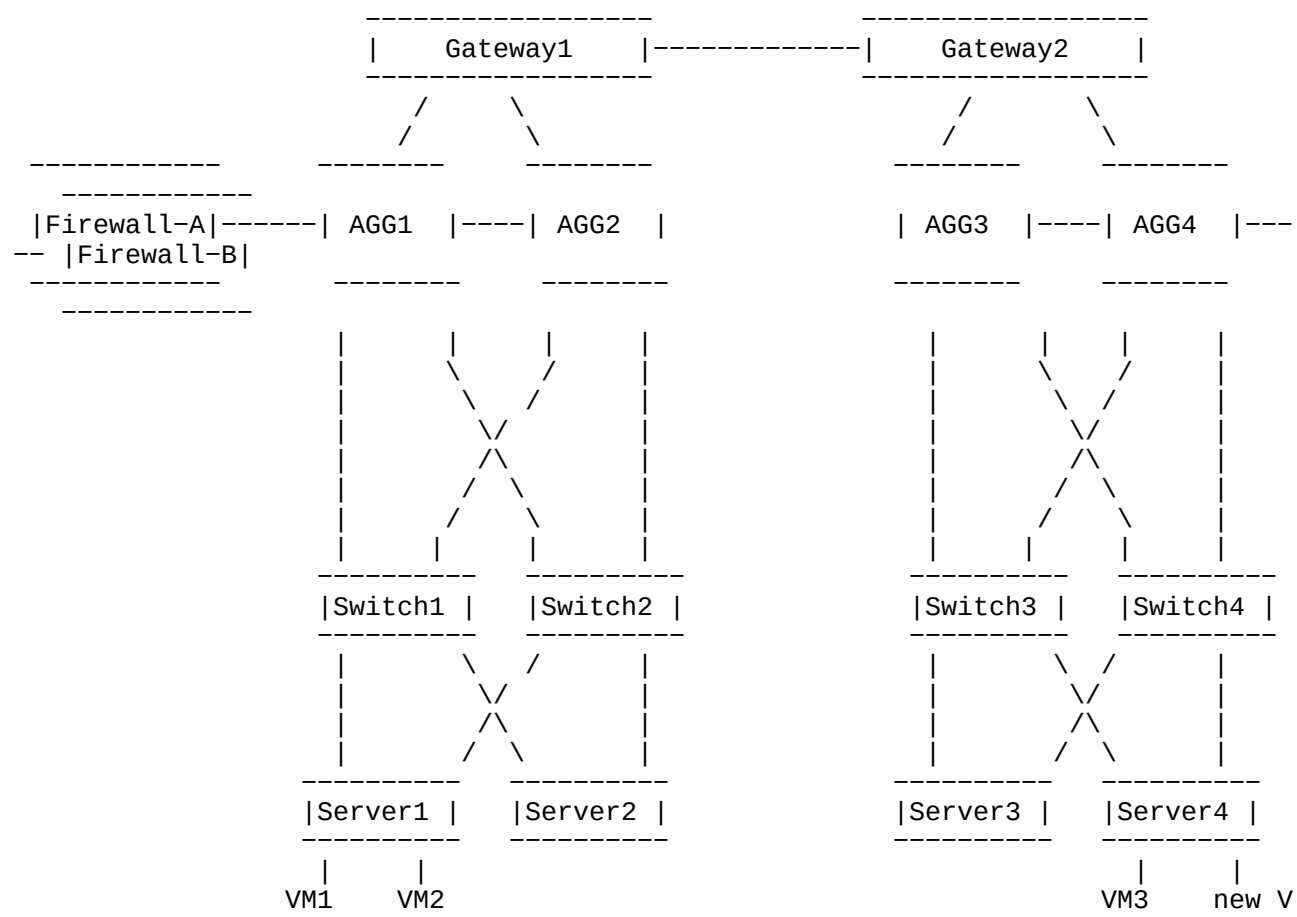

Fig1. Basic networking for discussion in this draft.

3.1. Policies and configurations

SLA is parsed into a collection of polices, which can be described by natural languages or mathematics fomula. Then policies are represented by specific configurations on different network elements, e.g. physical ports on routers and switches. In this draft, we discuss the migration of policies, but the policies migration also implies the migration of configurations on network devices, because configurations are embodiment of policies.

Policies that need to migrate with VM are those can influence VM's running services. The policies could be different on different network devices. For example, it can be static Access Control Lists on Access switches, QoS on switches and routers, security rules on Firewalls, etc.

Take Access Control List as an example. Figure 1 shows the influence

of lack of ACLs on destination switch. There is an ACL 100 on source switch (Switch1) deny all packets from IP subnet 10.138.3.0 to Internet. And another ACL 101 allows IP Address 10.138.3.1, VM1's IP Address, to send packets to Internet. VM1 has a running service on it. During service provisioning, VM1 is migrated to Server4 under Switch4, Where there is no ACL 100 and ACL 101. VM1's IP Address falls into a default ACL which deny all unmatching packets. As a result, packets belonging to the running services are dropped, hence the running service is interrupted.

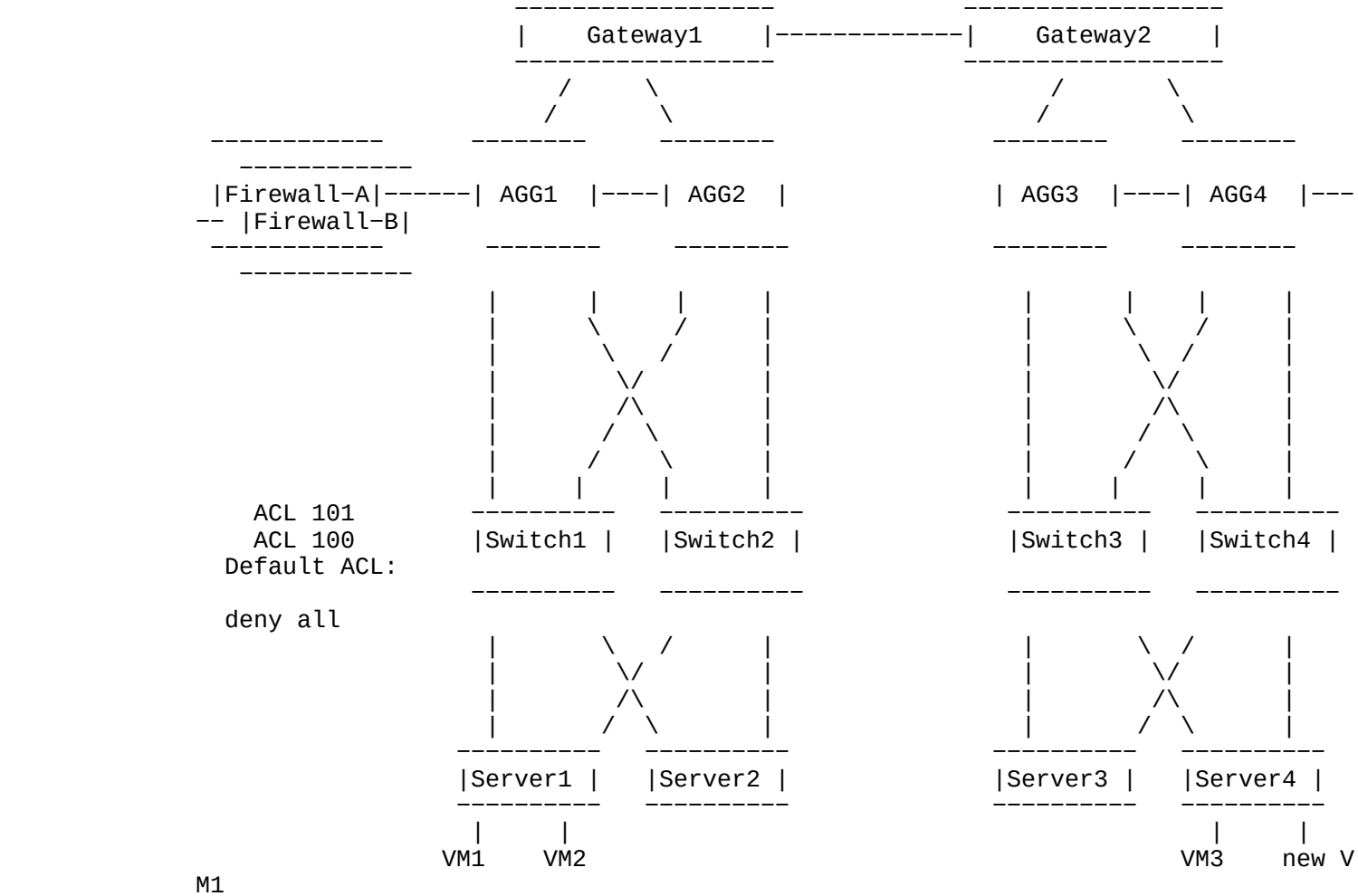

Fig.2 VM migration without ACL migration

4. Dynamic Information and the influence if lack or unaccurate

Network Manager (NM) can configure static configuration on network devices. Except for the static configuration, some dynamic information could also be recorded and processed by network devices. TCP connection table is an obvious example. Normally, TCP Connection Table is not configured by NM, but is generated by network devices, e.g. Firewalls, by looking into the packets passing them. TCP

Connection Table can be used for forwarding and security reasons. Another example is cumulated data, e.g. how many packets/TCP connecition requests an end host has sent. This information can only be generated by network devices according to real traffic. Configurations could be generated dynamically by network devices themselves according to the dynamic informaiton, e.g. Dynamic ACLs.

4.1. TCP connection tables

A typical TCP Connection Table includes the following data:

tcpConnState: The state of this TCP connection.

tcpConnLocalAddress: The local IP address for this TCP connection.

tcpConnLocalPort: The local port number for this TCP connection.

tcpConnRemAddress: The remote IP address for this TCP connection.

tcpConnRemPort: The remote port number for this TCP connection.

A TCP Connectin Table could also includes the following information:

Sequence Number: the sequence number in the packet header the sender is going to send.

Acknowledgement Number: the sequence number in the packet header the receiver is hoping to receive.

Idle time: the time that the tcp connection table hasn't been updated.

4.1.1. If TCP Connection Table isn't migrated

Assuming TCP Connection Table item is generated for VM1 on Firewall-A, the information is as follows:

tcpConnState == Established

tcpConnLocalAddress == 10.138.3.1

tcpConnLocalPort == 1234

tcpConnRemAddress: == 192.167.22.3

tcpConnRemPort == 4321

Assuming VM1 is migrated to Server4 under Switch4, without TCP

Connection Table migration. In order to keep the running service uninterrupted, the IP Address of VM1 will keep unchanged. The packets belonging to this TCP Connection will continue coming, which will pass Firewall-B, instead of Firewall-A. Because there is no TCP Connection Table for VM1 on Firewall-B, the following packets belonging to the TCP Connection will be dropped, hence the running service is broken down.

4.1.2. If TCP Connection Table is not accurately migrated

VM migration needs a period to finish memory and register copy. Fig.3 shows the VMware VMotion process. There are three points we should pay attention to.

Pre-copy period: VM begins to prepare for migration. In this period, VM pre-copy memory state to the new VM on destination device. The original VM is still power on and service is still running, which means the memory and register could keep changing. The new VM is power-on.

VM not running period: The end phase of memory copy. In this period, original VM stop running service, the memory will not change. Original VM finish copying the rest changed memory and register to new VM. New VM is still power-on.

VM power-off point: After original VM receives the OK message from new VM, it turns off the power, and meanwhile the new memory starts to run.

We can see that it's unrealistic to make a 'zero delay' VM migration, because there is at least about 1 second period (VM not running period) when neither VM is running.

Assuming there is a NM can GET and SET dynamic information. The NM GET dynamic information at Time A, and finish SET at Time B. At Time A, the Sequence Number of VM1's TCP Connection is 99. After NM GET dynamic information, VM1's TCP connection keeps transferring packets and Sequence Number increase to 110, until VM not running period begins. During VM not running period, no TCP packet is acknowledged by VM1, so the Sequence Number is 110. At Time B, the destination Firewall is SET by Sequence Number 99. When new VM1 starts, the packets belonging to VM1's TCP connection comes to Firewall-B with Sequence Number 111. Since the receiving Sequence Number doesn't equal to the Acknowleadge Sequence Number of Firewall-B, this packet will be dropped and the running service is broken down.

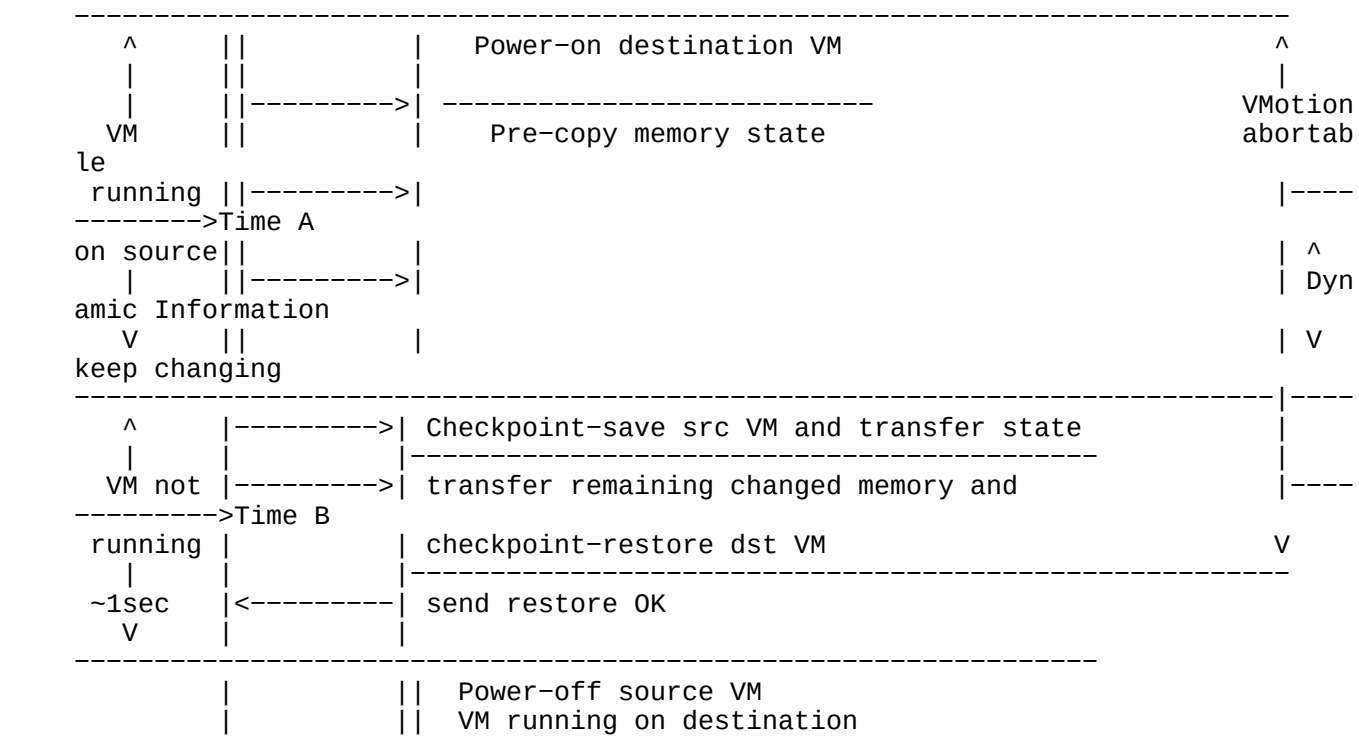

Fig.3 VMware VMotion process

4.2. Cumulated data

One example for cumulated data is unfinished TCP Connection established by a specific VM. In order to avoid TCP SYN flood, a network device may control the unfinished TCP Connection established by a single end host by setting a threshold. For example, the NM set the threshold to 5, and VM1 has established 3 unfinished TCP connections. If the cumulated TCP connection number isn't migrated to destination devices, the destination device will allow VM1 to establish up to 5 unfinished TCP connections. For the single destination device, the unfinished TCP connections established by VM1 is under control, but for the whole DC, VM1 has established 8 unfinished TCP connections. So VM1 has consume more resoureces than allowed.

4.3. Dynamic ACLs

Assuming all traffic is denied unless the end host is authorized and authenticated. VM1 has been authenticated on source device and a dynamic ACL has been generated to allow VM1's traffic to pass. If VM1 migrates to destination device without the dynamic ACL, the destination device will drop VM1's traffic, because VM1 is an unathenticated end host for it. So in this case, the dynamic ACL

needs to migrate with VM.

4.4. DHCP Snooping

Assuming source device is DHCP Snooping Enabled and a DHCP Snooping mapping item is created for VM1: (IP-VM1: MAC-VM1). This mapping is created dynamically by listening to DHCP Response message. If VM1 migrate to destination device, since the IP Address of VM1 doesn't change, there is no DHCP Request sent by VM1. So on destination device, there is (IP-VM1: MAC-VM1) mapping, all traffic from VM1 will be dropped. So DHCP Snooping mapping item need to migrate to destination device.

4.5. Multicast Membership

Multicast membership is similar to DHCP Snooping. Multicast membership is created on ports by listening to IGMP membership report messages. If VM1 migrates to destination, VM1 will not send IGMP membership report until next IGMP General Query. Before that, VM1 may not be able to recevie Multicast packets since network devices on and above destination devices don't knwo VM1's Multicast membership and don't forwarding the Multicast packets to VM1.

5. Existing network management protocol and the limitations

RFC3535 introduces many Network Management architectures and protocols. Basically, there are two kinds of architectures: network element oriented (SNMP and NETCONF) and Policy based (COPS-PR). In this section, we will introduce why these NMPs can not resolve the problem described in this draft.

5.1. Limitations

We analyze the problem described above into two aspects. One is Policies migration and the other is dynamic information migration.

5.1.1. For Policies migration

Existing NMP could be used to migrate Policies from source device to destination device. But we still need to face some questions:

Is device-oriented NMP suitable for policies migration?

Is C/S based NMP suitable for DC management?

How does NM know the source and destination device?

Do we need an automatic policies migration mechanism?

5.1.2. For Dynamic Information Migration

Currently, NMP is not used to configure dynamic information.

And, as Fig.3 shows, if we fail to begin migrating dynamic information at appropriate time (the time during VM not running period), the running services will be interrupted. For example, if dynamic information is migrated before 'VM not running period', dynamic information is inaccurate and the running services may be broken down when new VM restarts. In order to make accurate migration and keep running service uninterrupted, we need to know the exact timing for migration.

6. Security Considerations

The policies and dynamic information described above are all about security.

7. Acknowledgments

I would like to thank the following people for contributing to this draft: Ning Zong, David harrington, Linda dunbar, Susan Hares, Serge manning, Barry Leiba, Jiang xingfeng, Song Wei, Robert Sultan, and many others.

8. Normative Reference

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

Author's Address

Gu Yingjie
Huawei
No. 101 Software Avenue
Nanjing, Jiangsu Province 210001
P.R.China

Phone: +86-25-56624760
Fax: +86-25-56624702
Email: guyingjie@huawei.com

Operations Area Working Group
Internet-Draft
Intended status: Informational
Expires: July 27, 2013

T. Tsou
Huawei Technologies (USA)
J. Schoenwaelder, Ed.
Jacobs University Bremen
Y. Shi
T. Taylor
Huawei Technologies
G. Yang
China Telecom

January 23, 2013

Survey of Possibilities for the Automated Configuration of Large IP Networks
draft-ietf-opsawg-automated-network-configuration-05

Abstract

This memo discusses the steps required to bring a large number of devices into service in IP networks in an automated fashion. The goal of this document is to list known solutions where they exist, to point out approaches proven to be problematic, and to identify gaps that require further specifications.

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 http://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 July 27, 2013.

Table of Contents

1. Introduction

Many large IP networks are being deployed that entail the installation of tens of thousands of new network devices. To keep costs down, it is desirable to automate the establishment of such networks to the maximum extent possible. This naturally raises the question how new devices can pick up the configuration information they need to operate properly in an automated fashion. The goal of this document is to list known solutions where they exist, to point out approaches proven to be problematic, and to identify gaps that require further specifications.

The document primarily targets (a) network operators (in the generic sense) who are facing the challenge to roll out a large number of new devices and think about how to implement things properly, (b) network equipment vendors who like to add features to their products that make the roll out of lots of new devices simpler for their customers, and (c) people active in the IETF by identifying gaps where further standards may be useful to develop. The aim of the document is to provide guidance to actors who have not already experienced success in this area by informing about the trade-offs of different approaches.

A certain basic amount of configuration information must be pre-configured by the vendor or network operator before the devices are physically deployed. This pre-provisioned configuration can either be stored directly on the device itself or it can be provided to the device during the deployment operation via pluggable memory cards or near field communication technologies. Further device configuration information is best delivered after startup, to ensure that it is consistent with the physical deployment and the desired network configuration.

One example where automated configuration is important are new service provider networks. 3GPP work in progress describes requirements [TS_32_500] and an architectural specification [TS_36_300] for the self-configuration of edge node entities called eNodeBs. (The expansion of eNodeB is too unwieldy to spell out.) Specifically, procedures are specified for establishing transport connections to and for exchanging configuration data with control entities called MMEs (Mobility Management Entities) and with neighbouring eNodeBs. [TS_36_300] currently assumes as a starting precondition that the eNodeB knows its own IP address and knows IP address endpoints for the target MMEs and neighbouring eNodeBs.

The Broadband Forum has defined a CPE WAN Management Protocol (running over SOAP/HTTP/TLS) to manage customer premise equipment (CPE) terminating broadband access networks (typically DSL access

networks) [TR_069]. CPE devices locate and connect to an Auto-Configuration Server (ACS), which provides configuration data and software/firmware images and modules. The ACS also performs status and performance monitoring and diagnostic functions. CPE devices use DHCP to locate an ACS and since both peers, the ACS and CPE, can initiate connections, the protocol can work across network address translators (NATs). The DHCP exchange uses vendor-specific options defined by the Broadband Forum (number 3561 in the IANA Enterprise Numbers registry).

Next to service provider networks, many large enterprise networks face the same challenge to roll out a large number of network devices, which often connect to a 3rd party network provider. The current development of IP-based home automation and utility monitoring technologies might carry the problem to roll out large numbers of devices that need to automatically configure themselves to private households.

IETF work on automated configuration goes back to BOOTP [RFC0951], followed eight years later by DHCP ([RFC1541] and successors). The years since have seen a steady growth in the number of DHCP options. The Simple Network Management Protocol (SNMP) [RFC3410] was designed to convey management information between SNMP entities such as managers and agents. The number of SNMP MIB modules grew steadily, but SNMP has historically seen only limited use for configuration [RFC3535]. For a period, IETF configuration efforts were focussed on the distribution of policy information in the network. [RFC3139] provides a good insight into this period. More recently, the network configuration protocol NETCONF [RFC6241] was devised as an alternative to SNMP, but the development of standard NETCONF configuration data models is just beginning.

Recent IETF work closest in spirit to the 3GPP self-organizing network effort cited above is embodied in CAPWAP [RFC5415]. Like the 3GPP work, CAPWAP focusses on the configuration of edge nodes, in a Wi-Fi rather than cellular network. The CAPWAP work goes beyond that of 3GPP by specifying the process of Access Controller (AC) discovery rather than leaving discovery out of scope. A CAPWAP Wireless Termination Point (WTP) may use broadcasts and multicasts to discover local ACs, it may use CAPWAP DHCP options [RFC5417] to obtain IP addresses of ACs, or it may utilize CAPWAP DNS SRV records if a domain name is known. With regard to the configuration process itself, CAPWAP provides for the download of new images to the WTP (Wireless Termination Point). In contrast, [TS_32_500] assumes that this has already been completed for the eNodeB.

As can seen, standards for the automated configuration of devices in IP networks have so far been primarily developed for specific network

access technologies (3GPP, Broadband, 802.11 WLANs) and the various solutions make different assumptions about the services that are available and they are designed to support a configuration protocol that is specific to a certain access technology. The aim of this document is to analyse the various phases of an automated configuration process and to identify gaps that are currently not covered in standard and general purpose configuration management protocols of the IETF.

2. Intra-domain and Inter-domain Scenarios

There are two different scenarios to consider. In the first scenario, called the Intra-domain Scenario, the new network device N is attached to the network operated by the service provider which is also operating the new device. In the second scenario, called the Inter-domain Scenario, the new device N is attached to a third party network providing connectivity to the network of the service provider operating the new device.

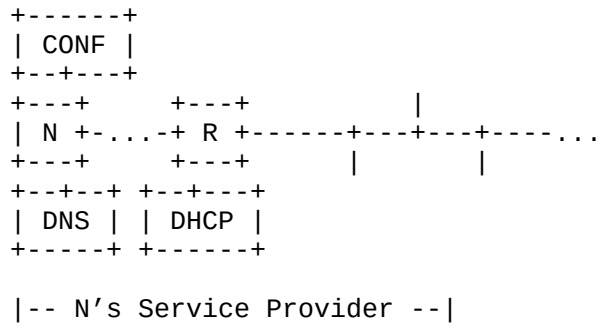

Figure 1: Intra-domain Scenario

Figure 1 depicts the Intra-domain Scenario. We assume that the new device N attaches to a link connected to router R. Furthermore, we assume that the service provider provides a Domain Name System (DNS) server, a reachable DHCP server, and a Configuration Server (CONF). Overall, this scenario does not differ much from conventional network scenarios.

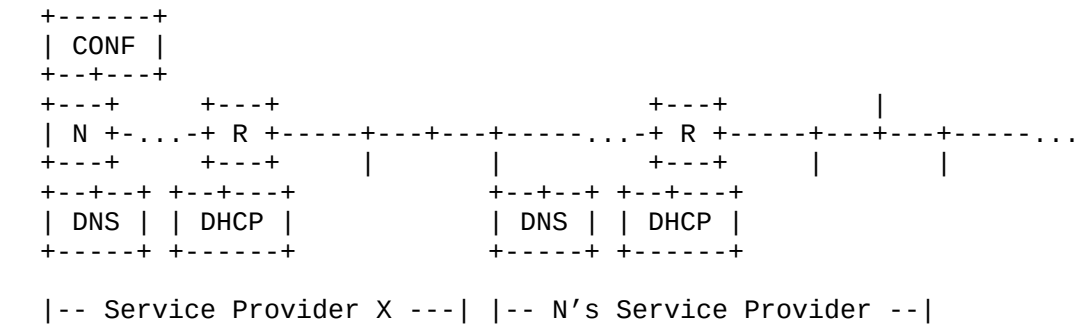

Figure 2: Inter-domain Scenario

Figure 2 depicts the Inter-domain Scenario where the new device N attaches to a router R owned by a different service provider X. The service provider X might offer its own DNS service and a reachable DHCP service. We assume that the service provider X has connectivity to the service provider planning to operate the new device.

It should be noted that handing out DHCP options specific to N's service provider via X's DHCP service requires some close coordination between the two parties involved. This might be difficult in practice. A more general alternative might be to have X's service provider establish a tunnel such that the new device logically appears to be part of N's service provider network.

In both scenarios, the new device N is either directly reachable or it may be behind a middlebox such as a Network Address Translator (NAT) or a firewall. Middleboxes may impose restrictions on which party is able to initiate communication. As detailed in [I-D.kwatsen-reverse-ssh], it is often desirable to allow device-initiated connections.

3. Model of the Automated Configuration Process

We introduce a model of the configuration process in order to identify the parts that have well-known solutions. The remainder may be worth studying to see if the industry can agree on a solution.

Some basic terminology is needed for the discussion. Depending on the implementation, let us agree that "configuration data" consist of software and sets of configured parameters in some combination. This includes firmware, licenses, certificates, and other configuration data. Also, the system that provides the configuration data is called the "configuration server". Finally, the term "joining

device" is used to denote a network device that is in the process of being incorporated into the network.

Broadly speaking, the configuration process can be broken into five phases:

1. Pre-configuration: configuration carried out either by the vendor or by the service provider prior to physical installation. One possible example is the pre-configuration of certificates or licenses or specific firmware.

2. Bootstrapping: the portion of the process from the time that physical installation is complete until a secure connection is established between the joining device and the configuration server.

3. Initial configuration: downloading of the configuration data that the joining device needs to carry out its function in the network.

4. Configuration auditing: tracking image versions and configuration parameters for each network device and verifying that the installed configuration data matches the physical installation, the network plan, and the records of what data was downloaded. It is possible that an initial audit of the physical installation is done before initial configuration, so that the validity of the intended download can be verified.

5. Configuration update: transferring configuration data to a fully configured and operating device from time to time as the need arises.

4. Phase 1: Pre-configuration

This memo identifies a specific requirement for pre-configuration of an invariant device identity and authentication-related material in the form of pre-shared secrets or certificates. There is, as one alternative, also a requirement for pre-configuration of information that permits the joining device to discover the address of the configuration server.

Note that pre-configuration may be carried out on the joining device itself or it may be provided to the joining device during the deployment process via pluggable memory cards or nearfield communication.

5. Phase 2: Bootstrapping

[I-D.sarikaya-core-sbootstrapping] deals with the process of security bootstrapping, with particular emphasis on the requirements for highly resource-constrained devices. The document makes a distinction between a data channel, which is used during network operation, and a control channel, which is used during bootstrapping. While both channels can be the same physical channel, they can also be different (e.g., a wireless access point using an infrared control channel to receive bootstrapping information). The draft discusses a number of possible security bootstrapping protocols for resource constrained devices that can be executed in several bootstrapping rounds and can be adapted to the specific contexts in terms of the resources available within individual devices and for the network as a whole.

For network devices in service provider networks or large enterprise networks, bootstrapping consists of several stages:

1. establishment of link layer connectivity with neighbouring nodes;

2. acquisition of IP addresses and basic routing information;

3. discovery of the configuration server;

4. establishment of a secure channel to the configuration server.

Each of these stages is further discussed below.

5.1. Establishment of Link Layer Connectivity

The protocol aspects of this phase are out of scope, since it involves non-IETF protocols only. While some link-layer technologies may provide authentication and access control, this cannot be assumed to be available in the general case.

5.2. Acquisition of IP Addresses and Basic Routing Information

For IPv4, DHCPv4 [RFC2131] is widely deployed and the usual way to obtain an IPv4 address, the IPv4 address of a link- local router and the IPv4 address of a DNS server. For IPv6, a choice has to be made between stateful DHCPv6 [RFC3315] versus stateless DHCPv6 [RFC3736] combined with stateless address autoconfiguration [RFC4862]. In the latter case, DHCPv6 is needed to configure parameters such as DNS server addresses. A routing advertisement option to configure the IPv6 address of a DNS server as part of the stateless address autoconfiguration is defined in [RFC6106].

Some security protection is provided in this stage by using DHCP authentication [RFC3118]. However, security of the configuration process as a whole has to be assured by other means. This is discussed further below.

Currently the lack of a stable identifier for use in DHCPv6 messaging is an impediment to authentication of the joining device. [RFC6355] discusses the problems with the current DHCPv6 identifiers (DUIDs) and proposes a new form that could be a more stable alternative.

A joining device can also choose to use a pre-configured IP address, a pre-configured link-local router address and a pre- configured DNS server address. This pre-configuration may be hard wired into the device or provided by a pluggable memory card or nearfield communication. However, a static pre-configuration hard- wires assumption about the network a devices operates in and is therefore brittle and not recommended.

5.3. Finding the Configuration Server

Four alternatives are available for finding the configuration server:

- pre-configuration;
- DHCP configuration;
- Service Location Protocol [RFC2608]; or
- DNS service discovery using DNS SRV records [RFC2782].

Pre-configuration of an IP address is brittle and not recommended unless the IP address is used as an anycast address. In the case of an IP anycast address, the routing system will select one out of an anycast cluster of configuration servers the devices connects to. For this to work well, all configuration servers in the anycast cluster should provide the same configuration data.

The pre-configuration of a Uniform Resource Identifier (URI) or fully qualified domain name (FQDN) is a slightly better approach than pre-configuring non-anycast IP addresses since this allows for a limited dynamic mapping of the name to an IP address. One variant that has been suggested is to burn the URI of a vendor server into the device’s firmware along with a device identifier, and have that server redirect to the URI of the service provider’s configuration server based on the device identity. Such an approach requires that the device vendor’s redirection server is always reachable, that the device vendor offers such a redirection service for the lifetime of their devices and that service providers are able to update the URI

of the service provider's redirection server. Furthermore, this approach can lead to problems if certificates are used to authenticate the involved parties if a service provider tries to prevent the usage of a vendor's redirection service. Finally, this approach also requires a trust relationship between the vendor and the service provider and agreement on a protocol to update the redirect information on the vendor's server. As a consequence of these considerations, using this approach is not recommended.

DHCP configuration can use the usual DHCP options and is technically straightforward since DHCP is widely used by end user devices to obtain basic configuration information. There is, however, no standardized DHCP option to communicate the address of a configuration server.

The Service Location Protocol (SLP) has seen some usage to locate services such as printers or file system shares. Usage of SLP to locate configuration servers requires to define a new service template [RFC2609].

The use of DNS SRV records requires the joining device to obtain the correct domain suffix first, presumably from DHCP or via Routing Advertisements in the case of IPv6 or pre-configuration. A service type for the desired configuration protocol would have to be defined in the DNS for the purpose. See Section 3.3 of [RFC5415] for a discussion of the corresponding discovery process for CAPWAP.

The Inter-domain Scenario requires that the DHCP server or the SLP server of service provider X's network is able to provide the correct information to the joining devices. To accomplish this, the discovery servers need to be able to match a device identification against a list of possible configuration servers. Furthermore, there needs to be a mechanism for the service provider operating the joining device to provision the configuration server's address, e.g., by using an extension of the Extensible Provisioning Protocol (EPP) [RFC5730]. However, if the joining device has pre- configured information about the name of the service provider's network, DNS SRV records may be queried after obtaining IP connectivity, avoiding the need to provision information in service provider X's network.

5.4. Establishing a Secure Channel to the Configuration Server

It is essential that the configuration server and the joining device authenticate themselves to each other, since the steps leading up to this point in the process may not be fully secure. This raises two issues: how the joining device identifies itself, and how authentication takes place.

It seems best if the device has an invariant identity built in and accessible to whatever operating system is running on it. [RFC6355] provides such an identity in the form of a Universally Unique IDentifier (UUID). The vendor should make that identity available in a form that can be read and transferred into a database accessible to the configuration server along with the associated configuration data in advance of the bootstrapping stage (e.g., in bar-coded format on the device packaging).

Serial numbers may be used for identification purposes if UUIDs are not available. However, serial numbers often encode information such as model-numbers or manufacturing dates. Hence, it is not recommended to pass serial-numbers in the clear for security reasons. Similar precautions apply to Common Language Equipment Identifier (CLEI) codes that encode information about properties of the device.

This leaves the mutual authentication process itself. This has two aspects: the security protocol used to perform authentication, and initial keying methodology. The security protocol is tied together with the choice of configuration data transport, but the basic choices are:

- IP Security (IPsec) [RFC4301];
- Transport Layer Security (TLS) [RFC5246];
- Datagram Transport Layer Security (DTLS) [RFC6347];
- Secure Shell (SSH) [RFC4251], [RFC4252], [RFC4253], and [RFC4254]; and
- SNMPv3's User-based Security Model (USM) [RFC3414].

For initial keying methodology, the two basic choices are between pre-shared secrets and certificates. All of the security protocols listed above except USM support both methods. USM supports pre-shared secrets only.

The usual concern with pre-shared secrets is scalability. In the bootstrapping case, the scale of operation required is linear with the number of devices to be configured, so it would definitely be a feasible approach if connection to the configuration system were the only consideration. The most likely procedure would be for the secret to be configured in the device during pre-configuration and also captured in a database along with the device identity, for use by the configuration server.

The problem with the use of pre-shared secrets is that the device

needs to authenticate itself at an earlier stage, while it is establishing communications with its neighbours and acquiring IP addresses. It seems undesirable to use the same secret that is used to authenticate the device to the configuration server for that purpose as well, on the basic principle of limiting the potential damage from disclosure of a particular key.

This need for additional pre-shared secrets argues for consideration of certificates as an alternative. One issue for certificates is where the trust anchor resides. It seems logical that it should reside with the service provider rather than the vendor, to make it easy to install equipment from multiple vendors. On that basis, pre-configuration requires service provider input. On the other hand, if devices are drop-shipped to the destination from the vendor, having the trust anchor reside with the vendor might be acceptable as well.

CAPWAP (Section 2.4.4.3 of [RFC5415]) makes use of the Extended Key Usage (EKU) certificate extension [RFC5280] to distinguish certificates identifying the Access Controllers (i.e., the configuration servers in the CAPWAP case) from the Wireless Transfer Points (the configured devices in the CAPWAP case). Thought should be given to whether such distinctions are required in the general case of network device configuration.

CAPWAP (Section 12.8 of [RFC5415]) also discusses the use of the Common Name rather than SubjectAltName field of the certificate to carry device identity, due to lack of a Uniform Resource Name (URN) specification allowing the use of SubjectAltName to carry MAC addresses. This encoding of device identifiers in certifications needs to be investigated further if a new form of device unique identity is used, as discussed above.

Middleboxes such as NATs or firewalls may impose restriction on which party is able to initiate communication. In the common case of NATs in IPv4 access networks, communication can only be established from the device to the configuration server. Not all secure transports, in particular those where authentication is not symmetric, support this "call home" mode of operation. A recent proposal to reverse the establishment of the TCP connection for SSH can be found in [I-D.kwatsen-reverse-ssh].

6. Phase 3: Initial Configuration

As mentioned at the beginning, the configuration data being downloaded may be a combination of software/firmware and configuration parameters. Some of the data will be vendor-specific and not subject to standardization. It appears that there is a

continuing debate on whether the configuration data should be pushed to the joining device or whether the device should pull the configuration data from the configuration server. In the latter case, the device needs to know about the existence of the data and the path to reach it before it can act. One way to acquire this information is through DHCP. DHCPv4 has provided the necessary options from its beginnings, inheriting them from BOOTP. They have been recently added to DHCPv6 [RFC5970].

Protocols that can transport configuration data can be classified as follows: The first class consists of generic file transfer protocols that can carry configuration data serialized into configuration files. The second class consists of protocols that manipulate structured configuration data directly. The structure of the configuration data is defined by some data model.

In the first class, we find the following file transfer protocols:

- The File Transfer Protocol (FTP) [RFC0959] can be used to move files containing configuration data. It can be secured by running FTP over TLS [RFC4217].

- The Trivial File Transfer Protocol (TFTP) [RFC1350] has been used extensively to load boot images over the network. However, it does not provide security and the only option is to rely on IP layer security (IPsec).

- The Hypertext Transfer Protocol (HTTP) [RFC2616] can be used to transfer documents containing configuration data. It is commonly secured by running HTTP over TLS [RFC2817], [RFC2818].

- The SSH File Transfer Protocol (SFTP) [I-D.ietf-secsh-filexfer] provides roughly the same services as FTP but runs over SSH and thus utilizes the security services provided by SSH.

- UNIX utilities to transfer files such as RCP and SCP provide limited flexibility and they differ in their degree of integration with SSH.

- The Control And Provisioning of Wireless Access Points (CAPWAP) protocol [RFC5415] can be used to control the download of images. CAPWAP can be secured by running CAPWAP over DTLS.

In the second class, we find the following configuration protocols:

- Version 3 of the Simple Network Management Protocol (SNMPv3) [RFC3411] can be used to manipulate MIB objects and to carry event notifications. SNMPv3 has its own security protocol (USM)

[RFC3414] but can also run over the secure transports SSH [RFC5592], TLS, or DTLS [RFC6353].

- The Common Open Policy Service for Policy Provisioning protocol (COPS-PR) [RFC3084] was designed to provision structured policy information from a Policy Decision Point (PDP) to a Policy Enforcement Point (PEP). The COPS protocol [RFC2748] provides an integrity object that can achieve authentication, message integrity, and replay prevention. Optionally, COPS and COPS-PR can run over TLS.

- The NETCONF protocol [RFC6241] provides mechanisms to install, manipulate, and delete the configuration of network devices. A protocol extension provides an asynchronous event notification delivery mechanism [RFC5277]. NETCONF by default runs over SSH but can also run over transports secured by TLS.

- The Control And Provisioning of Wireless Access Points protocol (CAPWAP) [RFC5415] supports the discovery of so called Access Controller (AC) by Wireless Termination Points (WTPs) and the configuration of WTPs by an AC. While CAPWAP can be extended to configure other devices, its main focus are WTPs. The CAPWAP protocol is protected by using DTLS after the discovery phase.

Table 1 lists the protocols plus their basic properties while Table 2 lists the security options available for each protocol.

Transport	Data Transfer Model
FTP	Push or pull of (configuration) files
TFTP	Push or pull of (configuration) files
HTTP	Push or pull of (configuration) files
SFTP	Push or pull of (configuration) files
RCP	Push or pull of (configuration) files
SCP	Push or pull of (configuration) files
CAPWAP	AC pushes configuration parameters, WTP pulls software
SNMPv3	Push of structured configuration parameters, event notifications
COPS-PR	Push of structured policy information
NETCONF	Push of structured configuration data, event notifications

Table 1: Protocols for transporting configuration data

Transport	IPsec	TLS	DTLS	SSH	Other
FTP	+	+			
TFTP	+				
HTTP	+	+			
SFTP	+			+	
RCP	+				
SCP	+			+	
CAPWAP	+		+		
SNMPv3	+	+	+	+	USM
COPS-PR	+	+			
NETCONF	+	+		+	

Table 2: Security options for configuration transport protocols

SNMPv3, NETCONF, and COPS-PR carry structured data specified in pre-defined data models. SNMPv3 and COPS-PR have size limitations on the data objects and thus make the transport of larger software images difficult. NETCONF does not suffer from hard size restrictions and can in principle carry software images inline. However, there is currently no work in progress to standardize the transfer of software images over NETCONF. CAPWAP combines the functions of configuration parameter transport and software download. The parameter transport aspect lacks the generality offered by SNMP, NETCONF, and COPS-PR, since the parameters are specified within the protocol specification itself. The remaining transports are independent of the nature of the information being transferred.

7. Phase 4: Configuration Auditing

To complete the process, it must be possible to audit the configuration status of the device in some detail. This is likely to begin even before all the configuration data has been downloaded. For instance, configuration management may wish to collect basic information such as the MAC addresses of the device’s interfaces, the link-local addresses assigned to them, and similar information for the neighbours of the joining device.

SNMP and SNMP MIB modules are obviously one way to collect this information. NETCONF [RFC6241] is an alternative, but the necessary data models have to be defined. YANG modules for NETCONF [RFC6020] can be generated from existing SNMP MIB modules by translating the SNMP modules into YANG modules [RFC6643].

Another important auditing activity is the analysis of system events.

The SYSLOG protocol [RFC5424] is widely used for this purpose but SNMPv3 and NETCONF can ship event notifications as well. Translations of SNMP notifications into structured SYSLOG messages and vice versa do exist [RFC5675], [RFC5676]. NETCONF can carry SYSLOG content as well [RFC5277].

NETCONF provides generic notifications that help with tracking configuration changes [RFC6470]. Similar standardized configuration change notifications do not exist for SNMP or SYSLOG.

8. Phase 5: Configuration Update

Configuration updates can in principle be handled with the same protocol that delivered the initial configuration. However, in some deployments, the mechanism used for initial configuration might be different.

An advantage of NETCONF over SNMPv3 and CAPWAP in the context of configuration updates is the support of concurrent updates through explicit locking mechanisms and the support of network wide configuration change transactions through the confirmed commit capability.

9. Gap Analysis

This document discussed the automated configuration of devices in large IP networks. Several gaps were identified requiring further specification:

G1: Definition of a DHCP option to provide the IPv4/IPv6 address of a configuration server. Such an option allows a joining device to pickup the configuration server’s address as part of the DHCP exchange. This is particularly interesting for Intra-domain Scenarios.

G2: Definition of DNS SRV records for locating configuration servers. Using SRV records, a joining device can lookup the configuration server’s address in the DNS. This is particularly useful in an Inter-domain Scenario.

G3: Definition of a SLP template for discovering configuration servers. Such a template is useful only in environments where SLP is used also for other purposes.

G4: Definition of NETCONF data models to support the download /update of software images through NETCONF.

G5: Definition of NETCONF data models for collecting basic system information and integrity information (e.g., checksums of software images).

G6: Some management protocols lack a mechanisms for devices to initiate a secure communication channel with a management system ("call home").

10. Security Considerations

The security of a configuration management solution is of crucial importance. Section 6 discusses the security options of several protocols that might be used. The relevant protocol definitions should be consulted to learn more about the specific security aspects of the various protocols.

It should be noted that some steps in the described process, in particular the bootstrapping phase, may not be secure and it is thus important to verify the identity of the device and the identity of the configuration server when a secure connection to a configuration server is established. Usage of IPsec, which focuses on securing the IP layer, may not be sufficient for this.

During the choice of protocols, the available security mechanisms and the required key management infrastructures may play a major role in the selection of protocols. Easy integration into existing Authentication, Authorization and Accounting (AAA) infrastructures can significantly reduce the operational costs associated with the security management of the configuration system.

While [I-D.sarikaya-core-sbootstrapping] discusses security bootstrapping mechanisms in the context of constrained devices, many of the mechanisms are also applicable for bootstrapping security in normal devices.

Finally, [RFC6092] discusses security capabilities for customer premises equipment providing residential IPv6 Internet service.

11. IANA Considerations

This memo includes no request to IANA.

12. Acknowledgements

Thanks to Ronald Bonica, Mehmet Ersue, Wesley George, Yiu Lee, Christopher Liljenstolpe, Kent Watsen, and Cathy Zhou for their comments during the preparation of this memo.

13. Informative References

[I-D.ietf-secsh-filexfer]
Galbraith, J. and O. Saarenmaa, ""SSH File Transfer Protocol", draft-ietf-secsh-filexfer-13 (work in progress)", July 2006.

[I-D.kwatsen-reverse-ssh]
Watsen, K., ""Reverse Secure Shell (Reverse SSH)", draft-kwatsen-reverse-ssh-01 (work in progress)", June 2011.

[I-D.sarikaya-core-sbootstrapping]
Sarikaya, B., Ohba, Y., Moskowitz, R., Cao, Z., and R. Cragie, "Security Bootstrapping Solution for Resource-Constrained Devices" draft-sarikaya-core-sbootstrapping-05 (work in progress)", July 2012.

[RFC0951] Croft, B. and J. Gilmore, "Bootstrap Protocol", RFC 951, September 1985.

[RFC0959] Postel, J. and J. Reynolds, "File Transfer Protocol", STD 9, RFC 959, October 1985.

[RFC1350] Sollins, K., "The TFTP Protocol (Revision 2)", STD 33, RFC 1350, July 1992.

[RFC1541] Droms, R., "Dynamic Host Configuration Protocol", RFC 1541, October 1993.

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, March 1997.

[RFC2608] Guttman, E., Perkins, C., Veizades, J., and M. Day, "Service Location Protocol, Version 2", RFC 2608, June 1999.

[RFC2609] Guttman, E., Perkins, C., and J. Kempf, "Service Templates and Service: Schemes", RFC 2609, June 1999.

[RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H.,

Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999.

[RFC2748] Durham, D., Boyle, J., Cohen, R., Herzog, S., Rajan, R., and A. Sastry, "The COPS (Common Open Policy Service) Protocol", RFC 2748, January 2000.

[RFC2782] Gulbrandsen, A., Vixie, P., and L. Esibov, "A DNS RR for specifying the location of services (DNS SRV)", RFC 2782, February 2000.

[RFC2817] Khare, R. and S. Lawrence, "Upgrading to TLS Within HTTP/1.1", RFC 2817, May 2000.

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, May 2000.

[RFC3084] Chan, K., Seligson, J., Durham, D., Gai, S., McCloghrie, K., Herzog, S., Reichmeyer, F., Yavatkar, R., and A. Smith, "COPS Usage for Policy Provisioning (COPS-PR)", RFC 3084, March 2001.

[RFC3118] Droms, R. and W. Arbaugh, "Authentication for DHCP Messages", RFC 3118, June 2001.

[RFC3139] Sanchez, L., McCloghrie, K., and J. Saperia, "Requirements for Configuration Management of IP-based Networks", RFC 3139, June 2001.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

[RFC3410] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", RFC 3410, December 2002.

[RFC3411] Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks", STD 62, RFC 3411, December 2002.

[RFC3414] Blumenthal, U. and B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", STD 62, RFC 3414, December 2002.

[RFC3535] Schoenwaelder, J., "Overview of the 2002 IAB Network Management Workshop", RFC 3535, May 2003.

[RFC3736] Droms, R., "Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6", RFC 3736, April 2004.

[RFC4217] Ford-Hutchinson, P., "Securing FTP with TLS", RFC 4217, October 2005.

[RFC4251] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Protocol Architecture", RFC 4251, January 2006.

[RFC4252] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Authentication Protocol", RFC 4252, January 2006.

[RFC4253] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Transport Layer Protocol", RFC 4253, January 2006.

[RFC4254] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Connection Protocol", RFC 4254, January 2006.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

[RFC5277] Chisholm, S. and H. Trevino, "NETCONF Event Notifications", RFC 5277, July 2008.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, May 2008.

[RFC5415] Calhoun, P., Montemurro, M., and D. Stanley, "Control And Provisioning of Wireless Access Points (CAPWAP) Protocol Specification", RFC 5415, March 2009.

[RFC5417] Calhoun, P., "Control And Provisioning of Wireless Access Points (CAPWAP) Access Controller DHCP Option", RFC 5417, March 2009.

[RFC5424] Gerhards, R., "The Syslog Protocol", RFC 5424, March 2009.

[RFC5592] Harrington, D., Salowey, J., and W. Hardaker, "Secure Shell Transport Model for the Simple Network Management Protocol (SNMP)", RFC 5592, June 2009.

[RFC5675] Marinov, V. and J. Schoenwaelder, "Mapping Simple Network Management Protocol (SNMP) Notifications to SYSLOG Messages", RFC 5675, October 2009.

[RFC5676] Schoenwaelder, J., Clemm, A., and A. Karmakar, "Definitions of Managed Objects for Mapping SYSLOG Messages to Simple Network Management Protocol (SNMP) Notifications", RFC 5676, October 2009.

[RFC5730] Hollenbeck, S., "Extensible Provisioning Protocol (EPP)", STD 69, RFC 5730, August 2009.

[RFC5970] Huth, T., Freimann, J., Zimmer, V., and D. Thaler, "DHCPv6 Options for Network Boot", RFC 5970, September 2010.

[RFC6020] Bjorklund, M., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, October 2010.

[RFC6092] Woodyatt, J., "Recommended Simple Security Capabilities in Customer Premises Equipment (CPE) for Providing Residential IPv6 Internet Service", RFC 6092, January 2011.

[RFC6106] Jeong, J., Park, S., Beloeil, L., and S. Madanapalli, "IPv6 Router Advertisement Options for DNS Configuration", RFC 6106, November 2010.

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

[RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, January 2012.

[RFC6353] Hardaker, W., "Transport Layer Security (TLS) Transport Model for the Simple Network Management Protocol (SNMP)", RFC 6353, July 2011.

[RFC6355] Narten, T. and J. Johnson, "Definition of the UUID-Based DHCPv6 Unique Identifier (DUID-UUID)", RFC 6355, August 2011.

[RFC6470] Bierman, A., "Network Configuration Protocol (NETCONF) Base Notifications", RFC 6470, February 2012.

[RFC6643] Schoenwaelder, J., "Translation of Structure of Management Information Version 2 (SMIv2) MIB Modules to YANG

Modules", RFC 6643, July 2012.

[TR_069] Blackford, J., Ed., Kirksey, H., Ed., and W. Lupton, Ed., ""CPE WAN Management Protocol", Broadband Forum TR-069", November 2010.

[TS_32_500] 3GPP, ""3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Telecommunication Management; Self-Organizing Networks (SON); Concepts and requirements (Release 9)", 3GPP TS 32.500", 2010.

[TS_36_300] 3GPP, ""3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Overall description; Stage 2 (Release 9)", 3GPP TS 36.300", 2010.

Authors' Addresses

Tina Tsou
Huawei Technologies (USA)
2330 Central Expressway
Santa Clara, CA 95050
USA

Phone:
Email: tina.tsou.zouting@huawei.com

Juergen Schoenwaelder (editor)
Jacobs University Bremen
Campus Ring 1
Bremen 28759
Germany

Phone:
Email: j.schoenwaelder@jacobs-university.de

Yang Shi
Huawei Technologies
156, Beiqing Road, Zhongguancun, Haidian District
Beijing
P.R. China

Phone: +86 10 60614043
Email: shiyang1@huawei.com

Tom Taylor
Huawei Technologies
Ottawa, Ontario
Canada

Phone:
Email: tom.taylor.stds@gmail.com

Guoliang Yang
China Telecom
No. 109 Zhongshan Ave. (West), Tianhe District
Guangzhou,
P.R. China

Phone: +86 020 38639615
Email: iamyanggl@gmail.com

Network Working Group
Internet-Draft
Intended status: Informational
Expires: September 20, 2012

M. Ersue, Ed.
Nokia Siemens Networks
B. Claise
Cisco Systems, Inc.
March 19, 2012

An Overview of the IETF Network Management Standards
draft-ietf-opsawg-management-stds-07

Abstract

This document gives an overview of the IETF network management standards and summarizes existing and ongoing development of IETF standards-track network management protocols and data models. The document refers to other overview documents, where they exist and classifies the standards for easy orientation. The purpose of this document is on the one hand to help system developers and users to select appropriate standard management protocols and data models to address relevant management needs. On the other hand, the document can be used as an overview and guideline by other Standard Development Organizations or bodies planning to use IETF management technologies and data models. This document does not cover OAM technologies on the data-path, e.g. OAM of tunnels, MPLS-TP OAM, and Pseudowire as well as the corresponding management models.

Table of Contents

1. Introduction

1.1. Scope and Target Audience

This document gives an overview of the IETF network management standards and summarizes existing and ongoing development of IETF standards-track network management protocols and data models. The document refers to other overview documents where they exist and classifies the standards for easy orientation.

The target audience of the document is on the one hand IETF working groups, which aim to select appropriate standard management protocols and data models to address their needs concerning network management. On the other hand the document can be used as an overview and guideline by non-IETF Standard Development Organizations (SDO) planning to use IETF management technologies and data models for the realization of management applications. The document can be also used to initiate a discussion between the bodies with the goal to gather new requirements and to detect possible gaps. Finally, this document is directed to all interested parties, which seek to get an overview of the current set of the IETF network management protocols such as network administrators or newcomers to IETF.

Section 2 gives an overview of the IETF core network management standards with a special focus on Simple Network Management Protocol (SNMP), SYSLOG, IP Flow Information Export/Packet Sampling (IPFIX/ PSAMP), and Network Configuration (NETCONF). Section 3 discusses IETF management protocols and mechanisms with a specific focus, e.g. IP address management or IP performance management. Section 4 discusses IETF data models, such as MIB modules, IPFIX Information Elements, SYSLOG Structured Data Elements, and YANG modules designed to address specific set of management issues and provides two complementary overviews for the network management data models standardized at IETF. Section 4.1 focuses on a broader view of models classified into categories such as generic and infrastructure data models as well as data models matched to different layers. Where section 4.2 structures the data models following the management application view and maps them to the network management tasks fault, configuration, accounting, performance, and security management.

Appendix A guides the reader for the high-level selection of management standards. For this, the section classifies the protocols according to high-level criteria such as push versus pull mechanism, passive versus active monitoring, as well as categorizes the protocols concerning the network management task they address and their data model extensibility. If the reader is interested only in a subset of the IETF network management protocols and data models described in this document, Appendix A can be used as a dispatcher to

the corresponding chapter. Appendix B gives an overview of the new work on Energy Management at IETF.

This document mainly refers to Proposed, Draft or Internet Standard documents at IETF (see [RFCSEARCH]). As far as valuable Best Current Practice (BCP) documents are referenced. In exceptional cases and if the document provides substantial guideline for standard usage or fills an essential gap, Experimental and Informational RFCs are noticed and ongoing work is mentioned.

Information on active and concluded IETF working groups (e.g., their charters, published or currently active documents and mail archive) can be found at [IETF-WGS]).

Note that this document does not cover OAM technologies on the data-path including MPLS forwarding plane, and control plane protocols (e.g. OAM of tunnels, MPLS-TP OAM, and Pseudowire) as well as the corresponding management models and MIB modules. For a list of related work see Section 1.2 "Related Work".

1.2. Related Work

[RFC6272] "Internet Protocols for the Smart Grid" gives an overview and guidance on the key protocols of the Internet Protocol Suite. In analogy to [RFC6272] this document gives an overview of the IETF network management standards and its usage scenarios.

[RFC3535] "Overview of the 2002 IAB Network Management Workshop" documented strengths and weaknesses of some IETF management protocols. In choosing existing protocol solutions to meet the management requirements, it is recommended that these strengths and weaknesses be considered, even though some of the recommendations from the 2002 IAB workshop have become outdated, some have been standardized, and some are being worked on at the IETF.

[RFC5706] "Guidelines for Considering Operations and Management of New Protocols and Extensions" recommends working groups to consider operations and management needs, and then select appropriate management protocols and data models. This document can be used to ease surveying the IETF standards-track network management protocols and management data models.

[RFC4221] "Multiprotocol Label Switching (MPLS) Management Overview" describes the management architecture for MPLS and indicates the interrelationships between the different MIB modules used for MPLS network management, where [RFC6371] "Operations, Administration, and Maintenance Framework for MPLS-Based Transport Networks" describes the OAM Framework for MPLS-based Transport Networks.

[I-D.ietf-mpls-tp-oam-analysis] "An Overview of the OAM Tool Set for MPLS-based Transport Networks" provides an overview of the OAM toolset for MPLS-based Transport Networks including a brief summary of MPLS-TP OAM requirements and functions, and of generic mechanisms created in the MPLS data plane to allow the OAM packets run in-band and share their fate with data packets. The protocol definitions for each MPLS-TP OAM tools are defined in separate documents, which are referenced.

[I-D.ietf-opsawg-oam-overview] "An Overview of Operations, Administration, and Maintenance (OAM) Mechanisms" gives an overview of the OAM toolset for detecting and reporting connection failures or measurement of connection performance parameters.

[I-D.ietf-mpls-tp-mib-management-overview] "MPLS-TP MIB-based Management Overview" describes the MIB-based architecture for MPLS-TP, and indicates the interrelationships between different existing MIB modules that can be leveraged for MPLS-TP network management and identifies areas where additional MIB modules are required.

Note that IETF so far has not developed specific technologies for the management of sensor networks. IP-based sensors or constrained devices in such an environment, i.e. with very limited memory and CPU resources, can use e.g. application layer protocols to do simple resource management and monitoring.

1.3. Terminology

This document does not describe standard requirements. Therefore, key words from RFC2119 are not used in the document.

- 3GPP: 3rd Generation Partnership Project, a collaboration between groups of telecommunications associations, to prepare the third-generation (3G) mobile phone system specification.

- Agent: A software module that performs the network management functions requested by network management stations. An agent may be implemented in any network element that is to be managed, such as a host, bridge, or router. The ’management server’ in NETCONF terminology.

- BCP: An IETF Best Current Practice document.

- CLI: Command Line Interface. A management interface that system administrators can use to interact with networking equipment.

- Data model: A mapping of the contents of an information model into a form that is specific to a particular type of data store or repository (see [RFC3444]).

- Event: An occurrence of something in the "real world". Events can be indicated to managers through an event message or notification.

- IAB: Internet Architecture Board

- IANA: Internet Assigned Numbers Authority, an organization that oversees global IP address allocation, autonomous system number allocation, media types, and other Internet Protocol-related code point allocations.

- Information model: An abstraction and representation of entities in a managed environment, their properties, attributes and operations, and the way they relate to each other. Independent of any specific repository, protocol, or platform (see [RFC3444]).

- ITU-T: International Telecommunication Union - Telecommunication Standardization Sector

- Managed object: A management abstraction of a resource; a piece of management information in a MIB module. In the context of SNMP, a structured set of data variables that represent some resource to be managed or other aspect of a managed device.

- Manager: An entity that acts in a manager role, either a user or an application. The counterpart to an agent. A 'management client' in NETCONF terminology.

- Management Information Base (MIB): An information repository with a collection of related objects that represent the resources to be managed.

- MIB module: MIB modules usually contain object definitions, may contain definitions of event notifications, and sometimes include compliance statements in terms of appropriate object and event notification groups. A MIB that is provided by a management agent is typically composed of multiple instantiated MIB modules.

- Modeling language: A modeling language is any artificial language that can be used to express information or knowledge or systems in a structure that is defined by a consistent set of rules. Examples are SMIv2 [STD58], XSD [XSD-1], and YANG [RFC6020].

- Notification: An unsolicited message sent by an agent to a management station to notify an unusual event.

- OAM: Operations, Administration, and Maintenance
- PDU: Protocol Data Unit, a unit of data, which is specified in a protocol of a given layer consisting protocol-control information and possibly layer-specific data.
- Principal: An application, an individual, or a set of individuals acting in a particular role, on whose behalf access to a service or MIB is allowed.
- Relax NG: REgular LAnguage for XML Next Generation, a schema language for XML [RELAX-NG].
- SDO: Standard Development Organization
- SMI: Structure of Managed Information, the notation and grammar for managed information definition used to define MIB modules [STD58].
- STDnn: An Internet Standard published at IETF, also referred as Standard, e.g. [STD62].
- URI: Uniform Resource Identifier, a string of characters used to identify a name or a resource on the Internet [STD66]. Can be classified as locators (URLs), or as names (URNs), or as both.
- XPATH: XML Path Language, a query language for selecting nodes from an XML document [XPATH].

2. Core Network Management Protocols

2.1. Simple Network Management Protocol (SNMP)

2.1.1. Architectural Principles of SNMP

The SNMPv3 Framework [RFC3410], builds upon both the original SNMPv1 and SNMPv2 framework. The basic structure and components for the SNMP framework did not change between its versions and comprises following components:

- managed nodes, each with an SNMP entity providing remote access to management instrumentation (the agent),
- at least one SNMP entity with management applications (the manager), and
- a management protocol used to convey management information between the SNMP entities, and management information.

During its evolution, the fundamental architecture of the SNMP Management Framework remained consistent based on a modular architecture, which consists of:

- o a generic protocol definition independent of the data it is carrying, and
- o a protocol-independent data definition language,
- o an information repository containing a data set of management information definitions (the Management Information Base, or MIB), and
- o security and administration.

As such following standards build up the basis of the current SNMP Management Framework:

- o SNMPv3 protocol [STD62],
- o the modeling language SMIv2 [STD58], and
- o MIB modules for different management issues.

The SNMPv3 Framework extends the architectural principles of SNMPv1 and SNMPv2 by:

- o building on these three basic architectural components, in some cases incorporating them from the SNMPv2 Framework by reference, and
- o by using the same layering principles in the definition of new capabilities in the security and administration portion of the architecture.

2.1.2. SNMP and its Versions

SNMP is based on three conceptual entities: Manager, Agent, and the Management Information Base (MIB). In any configuration, at least one manager node runs SNMP management software. Typically, network devices such as bridges, routers, and servers are equipped with an agent. The agent is responsible for providing access to a local MIB of objects that reflects the resources and activity at its node. Following the manager-agent paradigm, an agent can generate notifications and send them as unsolicited messages to the management application.

SNMPv2 enhances this basic functionality with an Inform PDU, a bulk

transfer capability and other functional extensions like an administrative model for access control, security extensions, and Manager-to-Manager communication. SNMPv2 entities can have a dual role as manager and agent. However, neither SNMPv1 nor SNMPv2 offers sufficient security features. To address the security deficiencies of SNMPv1/v2, SNMPv3 [STD62] has been issued.

[BCP74] "Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework" gives an overview of the relevant standard documents on the three SNMP versions. The BCP document furthermore describes how to convert MIB modules from SMIv1 to SMIv2 format and how to translate notification parameters as well as describes the mapping between the message processing and security models.

SNMP utilizes the Management Information Base, a virtual information store of modules of managed objects. Generally, standard MIB modules support common functionality in a device. Operators often define additional MIB modules for their enterprise or use the Command Line Interface (CLI) to configure non-standard data in managed devices and their interfaces.

SNMPv2 trap and inform PDUs can alert an operator or an application when some aspect of a protocol fails or encounters an error condition, and the contents of a notification can be used to guide subsequent SNMP polling to gather additional information about an event.

SNMP is widely used for monitoring of fault and performance data and with its stateless nature, SNMP also works well for status polling and determining the operational state of specific functionality. The widespread use of counters in standard MIB modules permits the interoperable comparison of statistics across devices from different vendors. Counters have been especially useful in monitoring bytes and packets going in and out over various protocol interfaces. SNMP is often used to poll basic parameter of a device (e.g. sysUpTime, which reports the time since the last re-initialization of the network management portion of the device) to check for operational liveliness, and to detect discontinuities in counters. Some operators use SNMP also for configuration management in their environment (e.g. for DOCSIS-based systems such as cable modems).

SNMPv1 [RFC1157] has been declared Historic and it is not recommended to use due to its lack of security features. [RFC1901] "Community-based SNMPv2" is an Experimental RFC, which has been declared Historic and it is not recommended to use due to its lack of security features.

SNMPv3 [STD62] is recommended to use due to its security features, including support for authentication, encryption, message timeliness and integrity checking, and fine-grained data access controls. An overview of the SNMPv3 document set is in [RFC3410].

Standards exist to use SNMP over diverse transport and link layer protocols, including Transmission Control Protocol (TCP) [STD7], User Datagram Protocol (UDP) [STD6], Ethernet [RFC4789], and others (see Section 2.1.5.1).

2.1.3. Structure of Managed Information (SMI)

SNMP MIB modules are defined with the notation and grammar specified as the Structure of Managed Information (SMI). The SMI uses an adapted subset of Abstract Syntax Notation One (ASN.1) [ITU-X680].

The SMI is divided into three parts: module definitions, object definitions, and, notification definitions.

- Module definitions are used when describing information modules. An ASN.1 macro, MODULE-IDENTITY, is used to concisely convey the semantics of an information module.

- Object definitions are used when describing managed objects. An ASN.1 macro, OBJECT-TYPE, is used to concisely convey the syntax and semantics of a managed object.

- Notification definitions are used when describing unsolicited transmissions of management information. An ASN.1 macro, NOTIFICATION-TYPE, is used to concisely convey the syntax and semantics of a notification.

SMIv1 is specified in [STD16][RFC1155] "Structure and Identification of Management Information for TCP/IP-based Internets" and [STD16][RFC1212] "Concise MIB Definitions". [RFC1215] specifies conventions for defining SNMP traps. Note that SMIv1 is outdated and is not recommended to use.

SMIv2 is the new notation for managed information definition and should be used to define MIB modules. SMIv2 is specified in following RFCs:

- [RFC2578], part of [STD58], defines Version 2 of the Structure of Management Information (SMIv2),

- [RFC2579], part of [STD58], defines the "Textual Conventions" macro for defining new types and it provides a core set of generally useful "Textual Convention" definitions,

- o [RFC2580], part of [STD58], defines Conformance Statements and requirements for defining agent and manager capabilities, and

- o [BCP74] defines the mapping rules for and the conversion of MIB modules between SMIv1 and SMIv2 formats.

2.1.4. SNMP Security and Access Control Models

2.1.4.1. Security Requirements on the SNMP Management Framework

Several of the classical threats to network protocols are applicable to management problem space and therefore applicable to any security model used in an SNMP Management Framework. This section lists primary and secondary threats, and threats which are of lesser importance (see [RFC3411] for the detailed description of the security threats).

The primary threats against which SNMP Security Models can provide protection are, "modification of information" by an unauthorized entity, and "masquerade", i.e. the danger that management operations not authorized for some principal may be attempted by assuming the identity of another principal.

Secondary threats against which SNMP Security Models can provide protection are "message stream modification", e.g. re-ordering, delay, or replay of messages, and "disclosure", i.e. the danger of eavesdropping on the exchanges between SNMP engines.

There are two threats against which SNMP Security Model does not protect, since they are deemed to be of lesser importance in this context: "Denial of Service" and "Traffic Analysis" (see [RFC3411]).

2.1.4.2. User-Based Security Model (USM)

SNMPv3 [STD62] introduced the User Security Model (USM). USM is specified in [RFC3414] and provides authentication and privacy services for SNMP. Specifically, USM is designed to secure against the primary and secondary threats discussed in Section 2.1.4.1. USM does not secure against Denial of Service and attacks based on Traffic Analysis.

The security services the USM security model supports are:

- o Data Integrity is the provision of the property that data has not been altered or destroyed in an unauthorized manner, nor have data sequences been altered to an extent greater than can occur non-maliciously.

- Data Origin Authentication is the provision of the property that the claimed identity of the user on whose behalf received data was originated is supported.

- Data Confidentiality is the provision of the property that information is not made available or disclosed to unauthorized individuals, entities, or processes.

- Message timeliness and limited replay protection is the provision of the property that a message whose generation time is outside of a specified time window is not accepted.

See [RFC3414] for a detailed description of SNMPv3 USM.

2.1.4.3. View-Based Access Control Model (VACM)

SNMPv3 [STD62] introduced the View-Based Access Control (VACM) facility. The VACM is defined in [RFC3415] and enables the configuration of agents to provide different levels of access to the agent's MIB. An agent entity can restrict access to a certain portion of its MIB, e.g. restrict some principals to view only performance-related statistics, or disallow other principals to read those performance-related statistics. An agent entity can also restrict the access to monitoring (read-only) as opposed to monitoring and configuration (read-write) of a certain portion of its MIB, e.g. allowing only a single designated principal to update configuration parameters.

VACM defines five elements that make up the Access Control Model: groups, security level, contexts, MIB views, and access policy. Access to a MIB module is controlled by means of a MIB view.

See [RFC3415] for a detailed description of SNMPv3 VACM.

2.1.5. SNMP Transport Subsystem and Transport Models

The User-based Security Model (USM) was designed to be independent of other existing security infrastructures to ensure it could function when third-party authentication services were not available. As a result, USM utilizes a separate user and key-management infrastructure. Operators have reported that the deployment of a separate user and key-management infrastructure in order to use SNMPv3 is costly and hinders the deployment of SNMPv3.

SNMP Transport Subsystem [RFC5590] extends the original SNMP architecture and transport model and enables the use of transport protocols to provide message security unifying the administrative security management for SNMP, and other management interfaces.

Transport Models are tied into the SNMP framework through the Transport Subsystem. The Transport Security Model [RFC5591] has been designed to work on top of lower-layer, secure Transport Models.

The SNMP Transport Model defines an alternative to existing standard transport mappings described in [RFC3417] e.g. for SNMP over UDP, in [RFC4789] for SNMP over IEEE 802 networks as well as in the Experimental RFC [RFC3430] defining SNMP over TCP.

2.1.5.1. SNMP Transport Security Model

The SNMP Transport Security Model [RFC5591] is an alternative to the existing SNMPv1 and SNMPv2 Community-based Security Models [BCP74], and the User-based Security Model [STD62][RFC3414].

The Transport Security Model utilizes one or more lower-layer security mechanisms to provide message-oriented security services. These include authentication of the sender, encryption, timeliness checking, and data integrity checking.

A secure transport model sets up an authenticated and possibly encrypted session between the Transport Models of two SNMP engines. After a transport-layer session is established, SNMP messages can be sent through this session from one SNMP engine to the other. The new Transport Model supports the sending of multiple SNMP messages through the same session to amortize the costs of establishing a security association.

The Secure Shell (SSH) Transport Model [RFC5592] and the Transport Layer Security (TLS) Transport Model [RFC6353] are current examples for Transport Security Models.

The SSH Transport Model makes use of the commonly deployed SSH security and key-management infrastructure. [RFC5592] furthermore defines MIB objects for monitoring and managing the SSH Transport Model for SNMP.

The Transport Layer Security (TLS) transport model [RFC6353] uses either the TLS protocol [RFC5246] or the Datagram Transport Layer Security (DTLS) [RFC6347] protocol. The TLS and DTLS protocols provide authentication and privacy services for SNMP applications. TLS transport model supports the sending of SNMP messages over TLS and TCP and over DTLS and UDP. [RFC6353] furthermore defines MIB objects for managing the TLS Transport Model for SNMP.

[RFC5608] describes the use of a 'Remote Authentication Dial-In User Service' (RADIUS) service by SNMP secure Transport Models for authentication of users and authorization of services. Access

control authorization, i.e. how RADIUS attributes and messages are applied to the specific application area of SNMP Access Control Models, and VACM in particular has been specified in [RFC6065].

2.2. SYSLOG Protocol

Syslog is a mechanism for distribution of logging information initially used on Unix systems (see [RFC3164] for BSD Syslog). The IETF SYSLOG protocol [RFC5424] introduces a layered architecture allowing the use of any number of transport protocols, including reliable and secure transports, for transmission of SYSLOG messages.

The SYSLOG protocol enables a machine to send system log messages across networks to event message collectors. The protocol is simply designed to transport and distribute these event messages. By default, no acknowledgements of the receipt are made, except the reliable delivery extensions specified in [RFC3195] are used. The SYSLOG protocol and process does not require a stringent coordination between the transport sender and the receiver. Indeed, the transmission of SYSLOG messages may be started on a device without a receiver being configured, or even actually physically present. Conversely, many devices will most likely be able to receive messages without explicit configuration or definitions.

BSD Syslog had little uniformity for the message format and the content of Syslog messages. The body of a BSD Syslog message has traditionally been unstructured text. This content is human-friendly, but difficult to parse for applications. The IETF has standardized a new message header format, including timestamp, hostname, application, and message ID, to improve filtering, interoperability and correlation between compliant implementations.

The SYSLOG protocol [RFC5424] introduces a mechanism for defining Structured Data Elements (SDEs). The SDEs allow vendors to define their own structured data elements to supplement standardized elements. [RFC5675] defines a mapping from SNMP notifications to SYSLOG messages. [RFC5676] defines a SNMP MIB module to represent SYSLOG messages for sending SYSLOG messages as notifications to SNMP notification receivers. [RFC5674] defines the way alarms are sent in SYSLOG, which includes the mapping of ITU perceived severities onto SYSLOG message fields and a number of alarm-specific definitions from ITU-T X.733 [ITU-X733] and the IETF Alarm MIB [RFC3877].

[RFC5848] "Signed Syslog Messages" defines a mechanism to add origin authentication, message integrity, replay resistance, message sequencing, and detection of missing messages to the transmitted SYSLOG messages to be used in conjunction with the SYSLOG protocol.

The SYSLOG protocol layered architecture provides support for a number of transport mappings. For interoperability purposes and especially in managed networks, where the network path has been explicitly provisioned for UDP syslog traffic, SYSLOG protocol can be used over UDP [RFC5426]. However, to support congestion control and reliability, [RFC5426] strongly recommends the use of the TLS transport.

[RFC3195] "Reliable Delivery for syslog" describes mappings of the SYSLOG protocol to TCP connections, useful for reliable delivery of event messages. As such the specification provides robustness and security in message delivery with encryption and authentication over a connection-oriented protocol that is unavailable to the usual UDP-based SYSLOG protocol.

IETF furthermore defined the TLS transport mapping for SYSLOG in [RFC5425], which provides a secure connection for the transport of SYSLOG messages. [RFC5425] describes the security threats to SYSLOG and how TLS can be used to counter such threats. [RFC6012] defines the Datagram Transport Layer Security (DTLS) Transport Mapping for SYSLOG, which can be used if a connectionless transport is desired.

For information on MIB modules related to SYSLOG see Section 4.2.1.

2.3. IP Flow Information Export (IPFIX) and Packet Sampling (PSAMP) Protocols

The IPFIX protocol [RFC5101], IP Flow Information eXport, defines a push-based data export mechanism for transferring IP flow information in a compact binary format from an exporter to a collector.

The IPFIX architecture [RFC5470] defines the components involved in IP flow measurement and reporting of information on IP flows, particularly, a metering process generating flow records, an exporting process that sends metered flow information using the IPFIX protocol, and a colleting process that receives flow information as IPFIX data records.

After listing the IPFIX requirements in [RFC3917], NetFlow Version 9 [RFC3954] was taken as the basis for the IPFIX protocol and the IPFIX architecture.

IPFIX can run over different transport protocols. The IPFIX protocol [RFC5101] specifies Stream Control Transmission Protocol (SCTP) [RFC4960] as the mandatory transport protocol to implement. Optional alternatives are TCP [STD7] and UDP [STD6].

SCTP is used with its Partial Reliability extension (PR-SCTP)

specified in [RFC3758]. [I-D.ietf-ipfix-export-per-sctp-stream] specifies an extension to [RFC5101], when using the PR-SCTP [RFC3758]. The extension offers several advantages over IPFIX export, e.g. the ability to calculate Data Record losses for PR-SCTP, immediate reuse of Template IDs within an SCTP stream, reduced likelihood of Data Record loss, and reduced demands on the Collecting Process.

IPFIX transmits IP flow information in data records containing IPFIX Information Elements (IEs) defined by the IPFIX information model [RFC5102]. IPFIX information elements are quantities with unit and semantics defined by the information model. When transmitted over the IPFIX protocol, only their values need to be carried in data records. This compact encoding allows efficient transport of large numbers of measured flow values. Remaining redundancy in data records can be further reduced by methods described in [RFC5473] (for further discussion on IPFIX IEs see Section 4).

The IPFIX information model is extensible. New information elements can be registered at IANA (see 'IPFIX Information Elements' in [IANA-PROT]). IPFIX also supports the use of proprietary, i.e. enterprise-specific information elements.

The PSAMP protocol [RFC5476] extends the IPFIX protocol by means of transferring information on individual packets. [RFC5475] specifies a set of sampling and filtering techniques for IP packet selection, based on the PSAMP framework [RFC5474]. The PSAMP information model [RFC5477] provides a set of basic information elements for reporting packet information with the IPFIX/PSAMP protocol.

The IPFIX model of an IP traffic flow is uni-directional. [RFC5103] adds means of reporting bi-directional flows to IPFIX, for example both directions of packet flows of a TCP connection.

When enterprise-specific information elements are transmitted with IPFIX, a collector receiving data records may not know the type of received data and cannot choose the right format for storing the contained information. [RFC5610] provides means of exporting extended type information for enterprise-specific Information Elements from an exporter to a collector.

Collectors may store received flow information in files. The IPFIX file format [RFC5655] can be used for storing IP flow information in a way that facilitates exchange of traffic flow information between different systems and applications.

In terms of IPFIX and PSAMP configurations, the metering and exporting processes are configured out of band. As the IPFIX

protocol is a push mechanism only, IPFIX cannot configure the exporter. The actual configuration of selection processes, caches, exporting processes, and collecting processes of IPFIX and PSAMP compliant monitoring devices is executed using the NETCONF protocol [RFC6241] (see Section 2.4.1). The 'Configuration Data Model for IPFIX and PSAMP' [I-D.ietf-ipfix-configuration-model] has been specified using Unified Modeling Language (UML) class diagrams. The data model is formally defined using the YANG modeling language [RFC6020] (see Section 2.4.2).

At the time of this writing a framework for IPFIX flow mediation is in preparation, which addresses the need for mediation of flow information in IPFIX applications in large operator networks, e.g. for aggregating huge amounts of flow data and for anonymization of flow information (see the problem statement in [RFC5982]).

The IPFIX Mediation Framework [RFC6183] defines the intermediate device between exporters and collectors, which provides an IPFIX mediation by receiving a record stream from e.g. a collecting process, hosting one or more intermediate processes to transform this stream, and exporting the transformed record stream into IPFIX messages via an exporting process.

Examples for mediation functions are flow aggregation, flow selection, and anonymization of traffic information (see [RFC6235]).

Privacy, integrity, and authentication of exporter and collector are important security requirements for IPFIX [RFC3917]. Confidentiality, integrity, and authenticity of IPFIX data transferred from an exporting process to a collecting process must be ensured. The IPFIX and PSAMP protocols do not define any new security mechanism and rely on the security mechanism of the underlying transport protocol, such as TLS [RFC5246] and DTLS [RFC6347].

The primary goal of IPFIX is the reporting of the flow accounting for flexible flow definitions and usage-based accounting. As described in the IPFIX Applicability Statement [RFC5472], there are also other applications such as traffic profiling, traffic engineering, intrusion detection, and QoS monitoring, that require flow-based traffic measurements and can be realized using IPFIX. IPFIX Applicability Statement explains furthermore the relation of IPFIX to other framework and protocols such as PSAMP, RMON (Remote Network Monitoring MIB Section 4.2.1), and IPPM (IP Performance Metrics Section 3.4)). Similar flow information could be also used for security monitoring. The addition of performance metrics in the IPFIX IANA registry [IANA-IPFIX], will extend the IPFIX use case to performance management.

Note that even if the initial IPFIX focus has been around IP flow information exchange, non-IP-related information elements are now specified in IPFIX IANA registration (e.g. MAC (Media Access Control) address, MPLS (Multiprotocol Label Switching) labels, etc.). At the time of this writing, there are requests to widen the focus of IPFIX and to export also non-IP related information elements (such as SIP monitoring IEs).

The IPFIX Structured Data [RFC6313] is an extension to the IPFIX protocol, which supports hierarchical structured data and lists (sequences) of Information Elements in data records. This extension allows the definition of complex data structures such as variable-length lists and specification of hierarchical containment relationships between templates. Furthermore, the extension provides the semantics to express the relationship among multiple list elements in a structured data record.

For information on data models related to the management of the IPFIX and PSAMP protocols see Section 4.2.1 and Section 4.2.2. For information on IPFIX/PSAMP IEs, see Section 4.2.3.

2.4. Network Configuration

2.4.1. Network Configuration Protocol (NETCONF)

The IAB workshop on Network Management [RFC3535] determined advanced requirements for configuration management:

- Robustness: Minimizing disruptions and maximizing stability,
- Support of task-oriented view,
- Extensible for new operations,
- Standardized error handling,
- Clear distinction between configuration data and operational state,
- Distribution of configurations to devices under transactional constraints,
- Single and multi-system transactions and scalability in the number of transactions and managed devices,
- Operations on selected subsets of management data,

- Dump and reload a device configuration in a textual format in a standard manner across multiple vendors and device types,
- Support a human interface and a programmatic interface,
- Data modeling language with a human friendly syntax,
- Easy conflict detection and configuration validation, and
- Secure transport, authentication, and robust access control.

The NETCONF protocol [RFC6241] provides mechanisms to install, manipulate, and delete the configuration of network devices and aims to address the configuration management requirements pointed in the IAB workshop. It uses an XML-based data encoding for the configuration data as well as the protocol messages. The NETCONF protocol operations are realized on top of a simple and reliable Remote Procedure Call (RPC) layer. A key aspect of NETCONF is that it allows the functionality of the management protocol to closely mirror the native command line interface of the device.

The NETCONF working group developed the NETCONF Event Notifications Mechanism as an optional capability, which provides an asynchronous message notification delivery service for NETCONF [RFC5277]. NETCONF notification mechanism enables using general purpose notification streams, where the originator of the notification stream can be any managed device (e.g. SNMP notifications).

NETCONF Partial Locking specification introduces fine-grained locking of the configuration datastore to enhance NETCONF for fine-grained transactions on parts of the datastore [RFC5717].

The NETCONF working group also defined the necessary data model to monitor the NETCONF protocol by using the modeling language YANG [RFC6022] (see Section 2.4.2). The monitoring data model includes information about NETCONF datastores, sessions, locks, and statistics, which facilitate the management of a NETCONF server.

NETCONF connections are required to provide authentication, data integrity, confidentiality, and replay protection. NETCONF depends on the underlying transport protocol for this capability. For example, connections can be encrypted in TLS or SSH, depending on the underlying protocol.

The NETCONF working group defined the SSH transport protocol as the mandatory transport binding [RFC6242]. Other optional transport bindings are TLS [RFC5539], BEEP (over TLS) [RFC4744], and SOAP (over HTTP over TLS) [RFC4743].

The NETCONF Access Control Model (NACM) [RFC6536] provides standard mechanisms to restrict protocol access to particular users with a pre-configured subset of operations and content.

2.4.2. YANG - NETCONF Data Modeling Language

Following the guidelines of the IAB management workshop [RFC3535], the NETMOD working group developed a data modeling language defining the semantics of operational and configuration data, notifications, and operations [RFC6020]. The new data modeling language maps directly to XML-encoded content (on the wire) and will serve as the normative description of NETCONF data models.

YANG has following properties addressing specific requirements on a modeling language for configuration management:

- YANG provides the means to define hierarchical data models. It supports reusable data types and groupings, i.e., a set of schema nodes that can be reused across module boundaries.

- YANG supports the distinction between configuration and state data. In addition, it provides support for modeling event notifications and the specification of operations that extend the base NETCONF operations.

- YANG allows to express constraints on data models by means of type restrictions and XPATH 1.0 [XPATH] expressions. XPATH expressions can also be used to make certain portions of a data model conditional.

- YANG supports the integration of standard and vendor defined data models. YANG's augmentation mechanism allows to seamlessly augment standard data models with proprietary extensions.

- YANG data models can be partitioned into collections of features, allowing low-end devices to only implement the core features of a data model while high-end devices may choose to support all features. The supported features are announced via the NETCONF capability exchange to management applications.

- The syntax of the YANG language is compact and optimized for human readers. An associated XML-based syntax called the YANG Independent Notation (YIN) [RFC6020] is available to allow the processing of YANG data models with XML-based tools. The mapping rules for the translation of YANG data models into Document Schema Definition Languages (DSDL), of which Relax NG is a major component, are defined in [RFC6110].

- Devices implementing standard data models can document deviations from the data model in separate YANG modules. Applications capable of discovering deviations can make allowances that would otherwise not be possible.

A collection of common data types for IETF-related standards is provided in [RFC6021]. This standard data type library has been derived to a large extend from common SMIv2 data types, generalizing them to a less constrained NETCONF framework.

The document "An Architecture for Network Management using NETCONF and YANG" describes how NETCONF and YANG can be used to build network management applications that meet the needs of network operators [RFC6244].

The Experimental RFC [RFC6095] specifies extensions for YANG introducing language abstractions such as class inheritance and recursive data structures.

[RFC6087] gives guidelines for the use of YANG within IETF and other standardization organizations.

Work is underway to standardize a translation of SMIv2 data models into YANG data models preserving investments into SNMP MIB modules, which are widely available for monitoring purposes.

Several independent and open source implementations of the YANG data modeling language and associated tools are available.

While YANG is a relatively recent data modeling language, some data models have already been produced. The specification of the base NETCONF protocol operations has been revised and uses YANG as the normative modeling language to specify its operations [RFC6241]. The IPFIX working group prepared the normative model for configuring and monitoring IPFIX and PSAMP compliant monitoring devices using the YANG modeling language [I-D.ietf-ipfix-configuration-model].

At the time of this writing the NETMOD working group is developing core system and interface data models. Following the example of the IPFIX configuration model, IETF working groups will prepare models for their specific needs.

For information on data models developed using the YANG modeling language see Section 4.2.1 and Section 4.2.2.

3. Network Management Protocols and Mechanisms with specific Focus

This section reviews additional protocols IETF offers for management and discusses for which applications they were designed and/or already successfully deployed. These are protocols that have mostly reached Proposed Standard status or higher within the IETF.

3.1. IP Address Management

3.1.1. Dynamic Host Configuration Protocol (DHCP)

Dynamic Host Configuration Protocol (DHCP) [RFC2131] provides a framework for passing configuration information to hosts on a TCP/IP network and enables as such auto-configuration in IP networks. In addition to IP address management, DHCP can also provide other configuration information, such as default routers, the IP addresses of recursive DNS servers and the IP addresses of NTP servers. As described in [RFC6272] DHCP can be used for IPv4 and IPv6 Address Allocation and Assignment as well as for Service Discovery.

There are two versions of DHCP, one for IPv4 (DHCPv4) [RFC2131] and one for IPv6 (DHCPv6) [RFC3315]. DHCPv4 was defined as an extension to BOOTP (Bootstrap Protocol) [RFC0951]. DHCPv6 was subsequently defined to accommodate new functions required by IPv6 such as assignment of multiple addresses to an interface and to address limitations in the design of DHCPv4 resulting from its origins in BOOTP. While both versions bear the same name and perform the same functionality, the details of DHCPv4 and DHCPv6 are sufficiently different that they can be considered separate protocols.

In addition to the assignment of IP addresses and other configuration information, DHCP options like the Relay Agent Information option (DHCPv4) [RFC3046] and, the Interface-Id Option (DHCPv6) [RFC3315] are widely used by ISPs.

DHCPv6 includes Prefix Delegation [RFC3633], which is used to provision a router with an IPv6 prefix for use in the subnetwork supported by the router.

Following are examples of DHCP options that provide configuration information or access to specific servers. A complete list of DHCP options is available at [IANA-PROT].

- [RFC3646] "DNS Configuration options for DHCPv6" describes DHCPv6 options for passing a list of available DNS recursive name servers and a domain search list to a client.

- o [RFC2610] "DHCP Options for Service Location Protocol" describes DHCPv4 options and methods through which entities using the Service Location Protocol can find out the address of Directory Agents in order to transact messages and how the assignment of scope for configuration of SLP User and Service Agents can be achieved.

- o [RFC3319] "DHCPv6 Options for Session Initiation Protocol (SIP) Servers" specifies DHCPv6 options that allow SIP clients to locate a local SIP server that is to be used for all outbound SIP requests, a so-called outbound proxy server.

- o [RFC4280] "DHCP Options for Broadcast and Multicast Control Servers" defines DHCPv6 options to discover the Broadcast and Multicast Service (BCMCS) controller in an IP network.

Built directly on UDP and IP, DHCP itself has no security provisions. There are two different classes of potential security issues related to DHCP: unauthorized DHCP Servers and unauthorized DHCP Clients. The recommended solutions to these risks generally involve providing security at lower layers, e.g. careful control over physical access to the network, security techniques implemented at layer two but also IPSec at layer three can be used to provide authentication.

3.1.2. Ad-Hoc Network Autoconfiguration

Ad-hoc nodes need to configure their network interfaces with locally unique addresses as well as globally routable IPv6 addresses, in order to communicate with devices on the Internet. The IETF AUTOCONF working group developed [RFC5889], which describes the addressing model for ad-hoc networks and how nodes in these networks configure their addresses.

The ad-hoc nodes under consideration are expected to be able to support multi-hop communication by running MANET (Mobile ad-hoc network) routing protocols as developed by the IETF MANET working group.

From the IP layer perspective, an ad hoc network presents itself as a layer 3 multi-hop network formed over a collection of links. The addressing model aims to avoid problems for ad-hoc-unaware parts of the system, such as standard applications running on an ad-hoc node or regular Internet nodes attached to the ad-hoc nodes.

3.2. IPv6 Network Operations

The IPv6 Operations Working Group develops guidelines for the operation of a shared IPv4/IPv6 Internet and provides operational

guidance on how to deploy IPv6 into existing IPv4-only networks, as well as into new network installations.

- [RFC4213] "Basic Transition Mechanisms for IPv6 Hosts and Routers" specifies IPv4 compatibility mechanisms for dual stack and configured tunneling that can be implemented by IPv6 hosts and routers. Dual stack implies providing complete implementations of both IPv4 and IPv6, and configured tunneling provides a means to carry IPv6 packets over unmodified IPv4 routing infrastructures.

- [RFC3574] "Transition Scenarios for 3GPP Networks" lists different scenarios in 3GPP defined packet network that would need IPv6 and IPv4 transition, where [RFC4215] "Analysis on IPv6 Transition in Third Generation Partnership Project (3GPP) Networks" does a more detailed analysis of the transition scenarios that may come up in the deployment phase of IPv6 in 3GPP packet networks.

- [RFC4029] "Scenarios and Analysis for Introducing IPv6 into ISP Networks" describes and analyzes different scenarios for the introduction of IPv6 into an ISP's existing IPv4 network. [RFC5181] "IPv6 Deployment Scenarios in 802.16 Networks" provides a detailed description of IPv6 deployment, integration methods and scenarios in wireless broadband access networks (802.16) in coexistence with deployed IPv4 services. [RFC4057] describes the scenarios for IPv6 deployment within enterprise networks.

- [RFC4038] "Application Aspects of IPv6 Transition" specifies scenarios and application aspects of IPv6 transition considering how to enable IPv6 support in applications running on IPv6 hosts, and giving guidance for the development of IP version-independent applications.

- The ongoing work on an IANA-reserved IPv4 prefix for shared address spaces [I-D.weil-shared-transition-space-request] "IANA Reserved IPv4 Prefix for Shared Address Space" updates RFC 5735 and requests the allocation of an IPv4/10 address block to be used as "Shared Carrier Grade Network Address Translation (CGN) Space" by service providers to number the interfaces that connect CGN devices to Customer Premise Equipment (CPE).

3.3. Policy-based Management

3.3.1. IETF Policy Framework

IETF specified a general policy framework [RFC2753] for managing, sharing, and reusing policies in a vendor independent, interoperable, and scalable manner. [RFC3460] specifies the Policy Core Information Model (PCIM) as an object-oriented information model for representing

policy information. PCIM has been developed jointly in the IETF Policy Framework working group and the Common Information Model (CIM) activity in the Distributed Management Task Force (DMTF). PCIM has been published as extensions to CIM [DMTF-CIM].

The IETF Policy Framework is based on a policy-based admission control specifying two main architectural elements, the Policy Enforcement Point (PEP) and the Policy Decision Point (PDP). For the purpose of network management, policies allow an operator to specify how the network is to be configured and monitored by using a descriptive language. Furthermore, it allows the automation of a number of management tasks, according to the requirements set out in the policy module.

IETF Policy Framework has been accepted by the industry as a standard-based policy management approach and has been adopted by different SDOs e.g. for 3GGP charging standards.

3.3.2. Use of Common Open Policy Service (COPS) for Policy Provisioning (COPS-PR)

[RFC3159] defines the Structure of Policy Provisioning Information (SPPI), an extension to the SMIv2 modeling language used to write Policy Information Base (PIB) modules. COPS-PR [RFC3084] uses the Common Open Policy Service (COPS) protocol [RFC2748] for provisioning of policy information. COPS provides a simple client/server model for supporting policy control over QoS signaling protocols. The COPS-PR specification is independent of the type of policy being provisioned (QoS, security, etc.) but focuses on the mechanisms and conventions used to communicate provisioned information between policy-decision-points (PDPs) and policy enforcement points (PEPs). Policy data is modeled using Policy Information Base (PIB) modules.

COPS-PR has not been widely deployed, and operators have stated that its use of binary encoding (BER) for management data makes it difficult to develop automated scripts for simple configuration management tasks in most text-based scripting languages. In the IAB Workshop on Network Management [RFC3535], the consensus of operators and protocol developers indicated a lack of interest in PIB modules for use with COPS-PR.

As a result, even if COPS-PR and the Structure of Policy Provisioning Information (SPPI) were initially approved as Proposed Standards, the IESG has not approved any PIB modules as IETF standard, and the use of COPS-PR is not recommended.

3.4. IP Performance Metrics (IPPM)

The IPPM working group has defined metrics for accurately measuring and reporting the quality, performance, and reliability of Internet data delivery. The metrics include connectivity, one-way delay and loss, round-trip delay and loss, delay variation, loss patterns, packet reordering, bulk transport capacity, and link bandwidth capacity.

These metrics are designed for use by network operators and their customers, and provide unbiased quantitative measures of performance. The IPPM metrics have been developed inside an active measurement context, that is, the devices used to measure the metrics produce their own traffic. However, most of the metrics can be used inside a passive context as well. At the time of this writing there is no work planned in the area of passive measurement.

As a property individual IPPM performance and reliability metrics need to be well-defined and concrete thus implementable. Furthermore, the methodology used to implement a metric needs to be repeatable with consistent measurements.

IETF IP Performance Metrics have been adopted by different organizations, e.g. Metro Ethernet Forum.

Note that this document does not aim to cover OAM technologies on the data-path and as such the discussion of IPPM-based active vs. passive monitoring as well as the data plane measurement and its diagnostics is rather incomplete. For a detailed overview and discussion of IETF OAM standards and IPPM measurement mechanisms the reader is referred to the documents listed at the end of Section 1.2 "Related Work" but especially to [I-D.ietf-opsawg-oam-overview].

Following are examples of essential IPPM documents:

- IPPM Framework document [RFC2330] defines a general framework for particular metrics developed by IPPM working group and defines the fundamental concepts of ’metric’ and ’measurement methodology’ and discusses the issue of measurement uncertainties and errors as well as introduces the notion of empirically defined metrics and how metrics can be composed.

- [RFC2679] "One-way Delay Metric for IPPM", defines a metric for one-way delay of packets across Internet paths. It builds on notions introduced in the IPPM Framework document.

- [RFC2681] "Round-trip Delay Metric for IPPM", defines a metric for round-trip delay of packets across network paths and follows

closely the corresponding metric for One-way Delay.

- [RFC3393] "IP Packet Delay Variation Metric", refers to a metric for variation in delay of packets across network paths and is based on the difference in the One-Way-Delay of selected packets called "IP Packet Delay Variation (ipdv)".

- [RFC2680] "One-way Packet Loss Metric for IPPM", defines a metric for one-way packet loss across Internet paths.

- [RFC5560] "One-Way Packet Duplication Metric", defines a metric for the case, where multiple copies of a packet are received and discusses methods to summarize the results of streams.

- [RFC4737] "Packet Reordering Metrics", defines metrics to evaluate whether a network has maintained packet order on a packet-by-packet basis and discusses the measurement issues, including the context information required for all metrics.

- [RFC2678] "IPPM Metrics for Measuring Connectivity", defines a series of metrics for connectivity between a pair of Internet hosts.

- [RFC5835] "Framework for Metric Composition", describes a detailed framework for composing and aggregating metrics.

- [BCP170] "Guidelines for Considering New Performance Metric Development" describes the framework and process for developing Performance Metrics of protocols and applications transported over IETF-specified protocols.

To measure these metrics two protocols and a sampling method have been standardized:

- [RFC4656] "A One-way Active Measurement Protocol (OWAMP)", measures unidirectional characteristics such as one-way delay and one-way loss between network devices and enables the interoperability of these measurements. OWAMP is discussed in more detail in [I-D.ietf-opsawg-oam-overview].

- [RFC5357] "A Two-Way Active Measurement Protocol (TWAMP)", adds round-trip or two-way measurement capabilities to OWAMP. TWAMP is discussed in more detail in [I-D.ietf-opsawg-oam-overview].

- [RFC3432] "Network performance measurement with Periodic Streams", describes a periodic sampling method and relevant metrics for assessing the performance of IP networks, as an alternative to the Poisson sampling method described in [RFC2330].

For information on MIB modules related to IP Performance Metrics see Section 4.2.4.

3.5. Remote Authentication Dial In User Service (RADIUS)

RADIUS [RFC2865], the Remote Authentication Dial In User Service, describes a client/server protocol for carrying authentication, authorization, and configuration information between a Network Access Server (NAS), which desires to authenticate its links and a shared Authentication Server. The companion document [RFC2866] 'Radius Accounting' describes a protocol for carrying accounting information between a network access server and a shared accounting server. [RFC2867] adds required new RADIUS accounting attributes and new values designed to support the provision of tunneling in dial-up networks.

The RADIUS protocol is widely used in environments like enterprise networks, where a single administrative authority manages the network, and protects the privacy of user information. RADIUS is deployed in fixed broadband access provider networks as well as in cellular broadband operators' networks.

RADIUS uses attributes to carry the specific authentication, authorization, information, and configuration details. RADIUS is extensible with a known limitation of maximum 255 attribute codes and 253 octets as attribute content length. RADIUS has Vendor-Specific Attributes (VSA), which have been used both for vendor-specific purposes as an addition to standardized attributes as well as to extend the limited attribute code space.

The RADIUS protocol uses a shared secret along with the MD5 (Message-Digest algorithm 5) hashing algorithm to secure passwords [RFC1321]. Based on the known threads additional protection like IPsec tunnels [RFC4301] are used to further protect the RADIUS traffic. However, building and administering large IPsec protected networks may become a management burden, especially when IPsec protected RADIUS infrastructure should provide inter-provider connectivity. A trend has been moving towards TLS-based security solutions [RFC5246] and establishing dynamic trust relationships between RADIUS servers. Since the introduction of TCP transport for RADIUS, it became natural to have TLS support for RADIUS. An ongoing work specifies the 'TLS encryption for RADIUS'.

[RFC2868] 'RADIUS Attributes for Tunnel Protocol Support' defines a number of RADIUS attributes designed to support the compulsory provision of tunneling in dial-up network access. Some applications involve compulsory tunneling i.e. the tunnel is created without any action from the user and without allowing the user any choice in the

matter. In order to provide this functionality, specific RADIUS attributes are needed to carry the tunneling information from the RADIUS server to the tunnel end points. [RFC3868] defines the necessary attributes, attribute values and the required IANA registries.

[RFC3162] 'RADIUS and IPv6' specifies the operation of RADIUS over IPv6 and the RADIUS attributes used to support the IPv6 network access. [RFC4818] describes how to transport delegated IPv6 prefix information over RADIUS.

[RFC4675] 'RADIUS Attributes for Virtual LAN and Priority Support' defines additional attributes for dynamic Virtual LAN assignment and prioritization, for use in provisioning of access to IEEE 802 local area networks usable with RADIUS and DIAMETER.

[RFC5080] 'Common RADIUS Implementation Issues and Suggested Fixes' describes common issues seen in RADIUS implementations and suggests some fixes. Where applicable, unclear statements and errors in previous RADIUS specifications are clarified. People designing extensions to RADIUS protocol for various deployment cases should get familiar with RADIUS Design Guidelines [RFC6158] in order to avoid e.g. known interoperability challenges.

[RFC5090] 'RADIUS Extension for Digest Authentication' defines an extension to the RADIUS protocol to enable support of Digest Authentication, for use with HTTP-style protocols like the Session Initiation Protocol (SIP) and HTTP.

[RFC5580] 'Carrying Location Objects in RADIUS and DIAMETER describes procedures for conveying access-network ownership and location information based on civic and geospatial location formats in RADIUS and DIAMETER.

[RFC5607] specifies required RADIUS attributes and their values for authorizing a management access to a NAS. Both local and remote management are supported, with access rights and management privileges. Specific provisions are made for remote management via Framed Management protocols, such as SNMP and NETCONF, and for management access over a secure transport protocols.

[RFC3579] describes how to use RADIUS to convey Extensible Authentication Protocol (EAP) [RFC3748] payload between the authenticator and the EAP server using RADIUS. RFC3579 is widely implemented, for example, in WLAN and 802.1 X environments. [RFC3580] describes how to use RADIUS with IEEE 802.1X authenticators. In the context of 802.1X and EAP-based authentication, the Vendor Specific Attributes described in [RFC2458]

have been widely accepted by the industry. [RFC2869] 'RADIUS extensions' is another important RFC related to EAP use. RFC2869 describes additional attributes for carrying AAA information between a NAS and a shared Accounting Server using RADIUS. It also defines attributes to encapsulate EAP message payload.

There are different MIB modules defined for multiple purposes to use with RADIUS (see Section 4.2.3 and Section 4.2.5).

3.6. Diameter Base Protocol (DIAMETER)

DIAMETER [RFC3588] provides an Authentication, Authorization and Accounting (AAA) framework for applications such as network access or IP mobility. DIAMETER is also intended to work in local AAA and in roaming scenarios. DIAMETER provides an upgrade path for RADIUS but is not directly backwards compatible.

DIAMETER is designed to resolve a number of known problems with RADIUS. DIAMETER supports server failover, reliable transport over TCP and SCTP, well documented functions for proxy, redirect and relay agent functions, server-initiated messages, auditability, and capability negotiation. DIAMETER also provides a larger attribute space for Attribute-Value Pairs (AVP) and identifiers than RADIUS. DIAMETER features make it especially appropriate for environments, where the providers of services are in different administrative domains than the maintainer (protector) of confidential user information.

Other notable differences to RADIUS are:

- Network and transport layer security (IPsec or TLS),
- Stateful and stateless models,
- Dynamic discovery of peers (using DNS SRV and NAPTR),
- Concept of an application that describes how a specific set of commands and Attribute-Value Pairs (AVPs) are treated by DIAMETER nodes. Each application has an IANA assigned unique identifier,
- Support of application layer acknowledgements, failover methods and state machines,
- Basic support for user-sessions and accounting,
- Better roaming support,

- Error notification, and
- Easy extensibility.

The DIAMETER protocol is designed to be extensible to support e.g. proxies, brokers, mobility and roaming, Network Access Servers (NASREQ), and Accounting and Resource Management. DIAMETER applications extend the DIAMETER base protocol by adding new commands and/or attributes. Each application is defined by a unique IANA assigned application identifier and can add new command codes and/or new mandatory AVPs.

The DIAMETER application identifier space has been divided into Standards Track and 'First Come First Served' vendor-specific applications. Following are examples for DIAMETER applications published at IETF:

- Diameter Base Protocol Application [RFC3588]: Required to support by all Diameter implementations.
- Diameter Base Accounting Application [RFC3588]: A DIAMETER application using an accounting protocol based on a server directed model with capabilities for real-time delivery of accounting information.
- Diameter Mobile IPv4 Application [RFC4004]: A DIAMETER application that allows a DIAMETER server to authenticate, authorize and collect accounting information for Mobile IPv4 services rendered to a mobile node.
- Diameter Network Access Server Application (NASREQ, [RFC4005]): A DIAMETER application used for AAA services in the NAS environment.
- Diameter Extensible Authentication Protocol Application [RFC4072]: A DIAMETER application that carries EAP packets between a NAS and a back-end authentication server.
- Diameter Credit-Control Application [RFC4006]: A DIAMETER application that can be used to implement real-time credit-control for a variety of end user services such as network access, Session Initiation Protocol (SIP) services, messaging services, and download services.
- Diameter Session Initiation Protocol Application [RFC4740]: A DIAMETER application designed to be used in conjunction with SIP and provides a DIAMETER client co-located with a SIP server, with the ability to request the authentication of users and authorization of SIP resources usage from a DIAMETER server.

- Diameter Quality-of-Service Application [RFC5866]: A DIAMETER application allowing network elements to interact with Diameter servers when allocating QoS resources in the network.

- Diameter Mobile IPv6 IKE (MIP6I) Application [RFC5778]: A DIAMETER application, which enables the interaction between a Mobile IP home agent and a Diameter server and is used when the mobile node is authenticated and authorized using IKEv2 [RFC5996].

- Diameter Mobile IPv6 Auth (MIP6A) Application [RFC5778]: A DIAMETER application, which enables the interaction between a Mobile IP home agent and a DIAMETER server and is used when the mobile node is authenticated and authorized using the Mobile IPv6 Authentication Protocol [RFC4285].

The large majority of DIAMETER applications are vendor-specific and mainly used in various SDOs outside IETF. One example SDO using DIAMETER extensively is 3GPP (e.g. 3GPP 'IP Multimedia Subsystem' (IMS) uses DIAMETER based interfaces (e.g. Cx) [3GPPIMS]). Recently, during the standardization of the '3GPP Evolved Packet Core' [3GPPEPC], DIAMETER was chosen as the only AAA signaling protocol.

One part of DIAMETER's extensibility mechanism is an easy and consistent way of creating new commands for the need of applications. RFC3588 proposed to define DIAMETER command code allocations with a new RFC. This policy decision caused undesired use and redefinition of existing Commands Codes within SDOs. Diverse RFCs have been published as simple command code allocations for other SDO purposes (see [RFC3589], [RFC5224], [RFC5431] and [RFC5516]). [RFC5719] changed the Command Code policy and added a range for vendor-specific Command Codes to be allocated on a 'First Come First Served' basis by IANA.

The implementation and deployment experience of DIAMETER has led to the currently ongoing development of an update of the base protocol [I-D.ietf-dime-rfc3588bis], which introduces TLS as the preferred security mechanism and deprecates the in-band security negotiation for TLS.

Some DIAMETER protocol enhancements and clarifications that logically fit better into [I-D.ietf-dime-rfc3588bis], are also needed on the existing RFC3588 based deployments. Therefore, protocol extensions specifically usable in large inter-provider roaming network scenarios are made available for RFC3588. Two currently existing specifications are mentioned below:

- o "Clarifications on the Routing of DIAMETER Requests Based on the Username and the Realm" [RFC5729] defines the behavior required for DIAMETER agents to route requests when the User-Name AVP contains a Network Access Identifier formatted with multiple realms. These multi-realm Network Access Identifiers are used in order to force the routing of request messages through a predefined list of mediating realms.

- o "Diameter Extended NAPTR" [RFC6408] describes an improved DNS-based dynamic DIAMETER Agent discovery mechanism without having to do DIAMETER capability exchange beforehand with a number of agents.

There have been a growing number of DIAMETER framework documents at IETF that basically are just a collection of AVPs for a specific purpose or a system architecture with semantical AVP descriptions and a logic for "imaginary" applications. From standardization point of view, this practice allows the development of larger system architecture documents that do not need to reference AVPs or application logic outside IETF. Below are examples of a few recent AVP and framework documents:

- o "Diameter Mobile IPv6: Support for Network Access Server to Diameter Server Interaction" [RFC5447] describes the bootstrapping of the Mobile IPv6 framework and the support of interworking with existing Authentication, Authorization, and Accounting (AAA) infrastructures by using the DIAMETER Network Access Server to home AAA server interface.

- o "Traffic Classification and Quality of Service (QoS) Attributes for Diameter" [RFC5777] defines a number of DIAMETER AVPs for traffic classification with actions for filtering and QoS treatment.

- o "Diameter Proxy Mobile IPv6: Mobile Access Gateway and Local Mobility Anchor Interaction with Diameter Server" [RFC5779] defines AAA interactions between Proxy Mobile IPv6 (PMIPv6) entities (Mobile Access Gateway and Local Mobility Anchor) and a AAA server within a PMIPv6 Domain.

For information on MIB modules related to DIAMETER see Section 4.2.5.

3.7. Control And Provisioning of Wireless Access Points (CAPWAP)

Wireless LAN (WLAN) product architectures have evolved from single autonomous Access Points to systems consisting of a centralized Access Controller (AC) and Wireless Termination Points (WTPs). The general goal of centralized control architectures is to move access

control, including user authentication and authorization, mobility management, and radio management from the single access point to a centralized controller, where an Access Points pulls the information from the Access Controller.

Based on the CAPWAP Architecture Taxonomy work [RFC4118] the CAPWAP working group developed the CAPWAP protocol [RFC5415] to facilitate control, management and provisioning of WTPs specifying the services, functions and resources relating to 802.11 WLAN Termination Points in order to allow for interoperable implementations of WTPs and ACs. The protocol defines the CAPWAP control plane including the primitives to control data access. The protocol document also specifies how configuration management of WTPs can be done and defines CAPWAP operations responsible for debugging, gathering statistics, logging, and firmware management as well as discusses operational and transport considerations.

The CAPWAP protocol is prepared to be independent of Layer 2 technologies, and meets the objectives in "Objectives for Control and Provisioning of Wireless Access Points (CAPWAP)" [RFC4564]. Separate binding extensions enable the use with additional wireless technologies. [RFC5416] defines CAPWAP Protocol Binding for IEEE 802.11.

CAPWAP Control messages, and optionally CAPWAP Data messages, are secured using DTLS [RFC6347]. DTLS is used as a tightly integrated, secure wrapper for the CAPWAP protocol.

For information on MIB modules related to CAPWAP see Section 4.2.2.

3.8. Access Node Control Protocol (ANCP)

The Access Node Control Protocol (ANCP) [RFC6320] realizes a control plane between a service-oriented layer 3 edge device, the Network Access Server (NAS) and a layer 2 Access Node (AN), e.g., Digital Subscriber Line Access Module (DSLAM). As such ANCP operates in a multi-service reference architecture and communicates QoS-, service- and subscriber-related configuration and operation information between a NAS and an Access Node.

The main goal of this protocol is to configure and manage access equipments and allow them to report information to the NAS in order to enable and optimize configuration.

The framework and requirements for an Access Node control mechanism and the use cases for ANCP are documented in [RFC5851].

The ANCP protocol offers authentication, and authorization between AN

and NAS nodes and provides replay protection and data-origin authentication. ANCP protocol solution is also robust against Denial-of-Service (DoS) attacks. Furthermore, the ANCP protocol solution is recommended to offer confidentiality protection. Security Threats and Security Requirements for ANCP are discussed in [RFC5713].

3.9. Application Configuration Access Protocol (ACAP)

The Application Configuration Access Protocol (ACAP) [RFC2244] is designed to support remote storage and access of program option, configuration and preference information. The data store model is designed to allow a client relatively simple access to interesting data, to allow new information to be easily added without server re-configuration, and to promote the use of both standardized data and custom or proprietary data. Key features include "inheritance" which can be used to manage default values for configuration settings and access control lists which allow interesting personal information to be shared and group information to be restricted.

ACAP’s primary purpose is to allow applications access to their configuration data from multiple network-connected computers. Users can then use any network-connected computer, run any ACAP-enabled application and have access to their own configuration data. To enable wide usage client simplicity has been preferred to server or protocol simplicity whenever reasonable.

The ACAP ’authenticate’ command uses Simple Authentication and Security Layer (SASL) [RFC4422] to provide basic authentication, authorization, integrity and privacy services. All ACAP implementations are required to implement the CRAM-MD5 (Challenge-Response Authentication Mechanism) [RFC2195] for authentication, which can be disabled based on the site security policy.

3.10. XML Configuration Access Protocol (XCAP)

The Extensible Markup Language (XML) Configuration Access Protocol (XCAP) [RFC4825] has been designed for and is commonly used with SIP-based solutions, in particular for instant message, presence, and SIP conference. XCAP is a protocol that allows a client to read, write, and modify application configuration data stored in XML format on a server, where the main functionality is provided by so called "XCAP Application Usages".

XCAP is a protocol that can be used to manipulate per-user data. XCAP is a set of conventions for mapping XML documents and document components into HTTP URIs, rules for how the modification of one resource affects another, data validation constraints, and

authorization policies associated with access to those resources. Because of this structure, normal HTTP primitives can be used to manipulate the data. Like ACAP, XCAP supports the configuration needs for a multiplicity of applications.

All XCAP servers are required to implement HTTP Digest Authentication [RFC2617]. Furthermore, XCAP servers are required to implement HTTP over TLS (HTTPS) [RFC2818]. It is recommended that administrators use an HTTPS URI as the XCAP root URI, so that the digest client authentication occurs over TLS.

Following list summarizes important XCAP application usages:

- o XCAP server capabilities [RFC4825] can be read by clients to determine which extensions, application usages, or namespaces a server supports.

- o A resource lists application is any application that needs access to a list of resources, identified by a URI, to which operations, such as subscriptions, can be applied [RFC4826].

- o A Resource List Server (RLS) Services application is a Session Initiation Protocol (SIP) application, where a server receives SIP SUBSCRIBE requests for resources, and generates subscriptions towards the resource list [RFC4826].

- o A Presence Rules application uses authorization policies, also known as authorization rules, to specify what presence information can be given to which watchers, and when [RFC4827].

- o A Pidf-manipulation application defines how XCAP is used to manipulate the contents of PIDF based presence documents [RFC4827].

4. Network Management Data Models

This section provides two complementary overviews for the network management data models standardized at IETF. The first subsection focuses on a broader view of models classified into categories such as generic and infrastructure data models as well as data models matched to different layers. The second subsection is structured following the management application view and focuses mainly on the data models for the network management tasks fault, configuration, accounting, performance, and security management (see [FCAPS]).

Note that IETF does not use the FCAPS view as an organizing principle for its data models. However, FCAPS view is used widely outside of IETF for the realization of management tasks and applications.

Section 4.2 aims to address the FCAPS view to enable people outside of IETF to understand the relevant data models at IETF.

The different data models covered in this section are MIB modules, IPFIX Information Elements, SYSLOG Structured Data Elements, and YANG modules. There are many technology-specific IETF data models, such as transmission and protocol MIBs, which are not mentioned in this document and can be found at [RFCSEARCH].

This section gives an overview of management data models that have reached Draft or Proposed Standard status at the IETF. In exceptional cases, important Informational RFCs are referred. The advancement process for management data models beyond Proposed Standard status, has been defined in [BCP27] with a more pragmatic approach and special considerations on data model specification interoperability. However, most IETF management data models never advanced beyond Proposed Standard.

4.1. IETF Network Management Data Models

The data models defined by the IETF can be broadly classified into the following categories depicted in Figure 1.

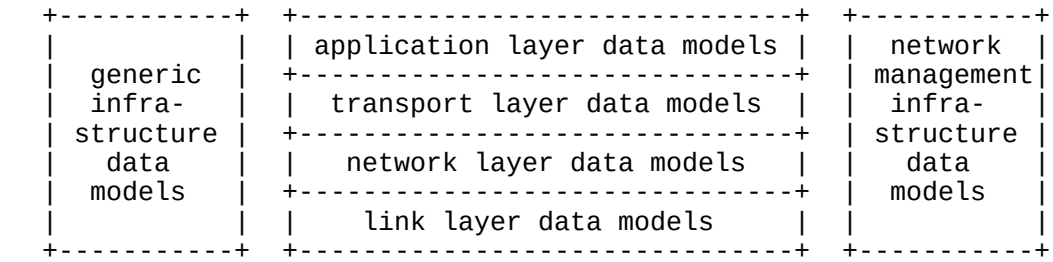

Figure 1: Categories of network management data models

Each of the categories is briefly described below. Note that the classification used here intends to provide orientation and reflects how most data models have been developed in the IETF by the various working groups. This classification does not aim to classify correctly all data models that have been defined by the IETF so far. The network layering model in the middle of Figure 1 follows the four layer model of the Internet as defined in [RFC1021].

The network management object identifiers for use with IETF MIB modules defined at IETF can be found under the IANA registry at [SMI-NUMBERS].

4.1.1. Generic Infrastructure Data Models

Generic infrastructure data models provide core abstractions that many other data models are built upon. The most important example is the interfaces data model defined in the IF-MIB [RFC2863]. It provides the basic notion of network interfaces and allows expressing stacking/layering relationships between interfaces. The interfaces data model also provides basic monitoring objects that are widely used for performance and fault management.

The second important infrastructure data model is defined in the Entity MIB [RFC4133]. It exports the containment hierarchy of the physical entities (slots, modules, ports) that make up a networking device and as such, it is a key data model for inventory management. Physical entities can have pointers to other data models that provide more specific information about them (e.g. physical ports usually point to the related network interface). Entity MIB extensions exist for physical sensors such as temperature sensors embedded on line cards or sensors that report fan rotation speeds [RFC3433]. Another extension models states and alarms of physical entities [RFC4268]. Some vendors have extended the basic Entity MIB with several proprietary data models.

4.1.2. Link Layer Data Models

A number of data models exist in the form of MIB modules covering the link layers IP runs over, such as ADSL [RFC4706], VDSL [RFC5650], GMPLS [RFC4803], ISDN [RFC2127], ATM [RFC2515] [RFC3606], Cable Modems [RFC4546] or Ethernet [RFC4188] [RFC4318] [RFC4363]. These so called transmission data models typically extend the generic network interfaces data model with interface type specific information. Most of the link layer data models focus on monitoring capabilities that can be used for performance and fault management functions and to some lesser extend for accounting and security management functions. The IEEE has meanwhile taken over the responsibility to maintain and further develop data models for the IEEE 802 family of protocols [RFC4663]. The cable modem industry consortium DOCSIS is working with the IETF to publish data models for cable modem networks as IETF standards-track specifications.

4.1.3. Network Layer Data Models

There are data models in the form of MIB modules covering IP/ICMP [RFC4293] [RFC4292] network protocols and their extensions (e.g., Mobile IP), the core protocols of the Internet. In addition, there are data models covering popular unicast routing protocols (OSPF [RFC4750], ISIS [RFC4444], BGP-4 [RFC4273]) and multicast routing protocols (PIM [RFC5060]).

Detailed models also exist for performance measurements in the form of IP performance metrics [RFC2330] (see Section 3.4).

The necessary data model infrastructure for configuration data models covering network layers are currently being defined using NETCONF [RFC6242] and YANG [RFC6020].

4.1.4. Transport Layer Data Models

There are data models for the transport protocols TCP [RFC4022], UDP [RFC4113], and SCTP [RFC3873]. For TCP, a data model providing extended statistics is defined in [RFC4898].

4.1.5. Application Layer Data Models

Some data models have been developed for specific application protocols (e.g., SIP [RFC4780]). In addition, there are data models that provide a generic infrastructure for instrumenting applications in order to obtain data useful primarily for performance management and fault management [RFC2287] [RFC2564]. In general, however, generic application MIB modules have been less successful in gaining widespread deployment.

4.1.6. Network Management Infrastructure Data Models

A number of data models are concerned with the network management system itself. This includes, in addition to a set of SNMP MIB modules for monitoring and configuring SNMP itself [RFC3410], some MIB modules providing generic functions such as the calculation of expressions over MIB objects, generic functions for thresholding and event generation, event notification logging functions and data models to represent alarms [RFC2981] [RFC2982] [RFC3014] [RFC3877]. In addition, there are data models that allow to execute basic reachability and path discovery tests [RFC4560]. Another collection of MIB modules provides remote monitoring functions, ranging from the data link layer up to the application layer. This is known as the RMON family of MIB modules [RFC3577].

The IPFIX protocol [RFC5101] (Section 2.3) is used to export information about network flows collected at so called observation points (typically a network interface). The information elements [RFC5102] carried in IPFIX cover the network and transport layer very well but also provides some link layer specific information elements. Work is underway to further extend the standardized information that can be carried in IPFIX.

The SYSLOG protocol document [RFC5424] (Section 2.2) defines an initial set of Structured Data Elements (SDEs) that relate to content

time quality, content origin, and meta-information about the message, such as language. Proprietary SDEs can be used to supplement the IETF- defined SDEs.

4.2. Network Management Data Models - FCAPS View

This subsection follows the management application view and aims to match the data models to network management tasks for fault, configuration, accounting, performance, and security management ([FCAPS]). As OAM is a general term that refers to a toolset, which can be used for fault detection, isolation, and performance measurement, aspects of FCAPS in the context of the data path, such as fault and performance management, are also discussed in [I-D.ietf-opsawg-oam-overview] "An Overview of Operations, Administration, and Maintenance (OAM) Mechanisms".

Some of the data models do not fit into one single FCAPS category per design but span multiple areas. For example, there are many technology-specific IETF data models, such as transmission and protocol MIBs, which cover multiple FCAPS categories, and therefore are not mentioned in this sub section and can be found at [RFCSEARCH].

4.2.1. Fault Management

Fault management encloses a set of functions to detect, isolate, notify, and correct faults encountered in a network as well as to maintain and examine error logs. The data models below can be utilized to realize a fault management application.

[RFC3418], part of SNMPv3 standard [STD62], is a MIB module containing objects in the system group that are often polled to determine if a device is still operating, and sysUpTime can be used to detect if the network management portion of the system has restarted, and counters have been reinitialized.

[RFC3413], part of SNMPv3 standard [STD62], is a MIB module including objects designed for managing notifications, including tables for addressing, retry parameters, security, lists of targets for notifications, and user customization filters.

The Interfaces Group MIB [RFC2863] builds on the old standard for MIB II [STD17] and is used as a primary MIB module for managing and monitoring the status of network interfaces. The Interfaces Group MIB defines a generic set of managed objects for network interfaces and it provides the infrastructure for additional managed objects specific to particular types of network interfaces, such as Ethernet.

[RFC4560] defines a MIB module for performing ping, traceroute, and lookup operations at a host. For troubleshooting purposes, it is useful to be able to initiate and retrieve the results of ping or traceroute operations when they are performed at a remote host.

The RMON (Remote Network Monitoring) MIB [STD59][RFC2819] can be configured to recognize conditions on existing MIB variables (most notably error conditions) and continuously to check for them. When one of these conditions occurs, the event may be logged, and management stations may be notified in a number of ways (for further discussion on RMON see Section 4.2.4).

DISMAN-EVENT-MIB in [RFC2981] and DISMAN-EXPRESSION-MIB in [RFC2982] provide a superset of the capabilities of the RMON alarm and event groups. These modules provide mechanisms for thresholding and reporting anomalous events to management applications.

The ALARM MIB in [RFC3877] and the Alarm Reporting Control MIB in [RFC3878] specify mechanisms for expressing state transition models for persistent problem states. ALARM MIB defines:
- a mechanism for expressing state transition models for persistent problem states,
- a mechanism to correlate a notification with subsequent state transition notifications about the same entity/object, and
- a generic alarm reporting mechanism (extends ITU-T work on X.733 [ITU-X733]).

[RFC3878] in particular defines objects for controlling the reporting of alarm conditions and extends ITU-T work on M.3100 Amendment 3 [ITU-M3100].

Other MIB modules that may be applied to fault management with SNMP include:

- NOTIFICATION-LOG-MIB [RFC3014] describes managed objects used for logging SNMP Notifications.

- ENTITY-STATE-MIB [RFC4268] describes extensions to the Entity MIB to provide information about the state of physical entities.

- ENTITY-SENSOR-MIB [RFC3433] describes managed objects for extending the Entity MIB to provide generalized access to information related to physical sensors, which are often found in networking equipment (such as chassis temperature, fan RPM, power supply voltage).

The SYSLOG protocol document [RFC5424] defines an initial set of Structured Data Elements (SDEs) that relate to content time quality,

content origin, and meta-information about the message, such as language. Proprietary SDEs can be used to supplement the IETF-defined SDEs.

The IETF has standardized MIB Textual-Conventions for facility and severity labels and codes to encourage consistency between SYSLOG and MIB representations of these event properties [RFC5427]. The intent is that these textual conventions will be imported and used in MIB modules that would otherwise define their own representations.

An IPFIX MIB module [RFC5815] has been defined for monitoring IPFIX meters, exporters and collectors (see Section 2.3). The ongoing work on PSAMP MIB module extends the IPFIX MIB modules by managed objects for monitoring PSAMP implementations [I-D.ietf-ipfix-psamp-mib].

The NETCONF working group defined the data model necessary to monitor the NETCONF protocol [RFC6022] with the modeling language YANG. The monitoring data model includes information about NETCONF datastores, sessions, locks, and statistics, which facilitate the management of a NETCONF server. NETCONF monitoring document also defines methods for NETCONF clients to discover the data models supported by a NETCONF server and defines the operation <get-schema> to retrieve them.

4.2.2. Configuration Management

Configuration management focuses on establishing and maintaining consistency of a system and defines the functionality to configure its functional and physical attributes as well as operational information throughout its life. Configuration management includes configuration of network devices, inventory management, and software management. The data models below can be used to utilize configuration management.

MIB modules for monitoring of network configuration (e.g. for physical and logical network topologies) already exist and provide some of the desired capabilities. New MIB modules might be developed for the target functionality to allow operators to monitor and modify the operational parameters, such as timer granularity, event reporting thresholds, target addresses, etc.

[RFC3418], part of [STD62], contains objects in the system group useful e.g. for identifying the type of device, and the location of the device, the person responsible for the device. The SNMPv3 standard [STD62] furthermore includes objects designed for configuring principals, access control rules, notification destinations, and for configuring proxy-forwarding SNMP agents, which can be used to forward messages through firewalls and Network Address Translation (NAT) devices.

The Entity MIB [RFC4133] supports mainly inventory management and is used for managing multiple logical and physical entities matched to a single SNMP agent. This module provides a useful mechanism for identifying the entities comprising a system and defines event notifications for configuration changes that may be useful to management applications.

[RFC3165] defines a set of managed objects that enable the delegation of management scripts to distributed managers.

For configuring IPFIX and PSMAP devices, the IPFIX working group developed the IPFIX configuration data model [I-D.ietf-ipfix-configuration-model], by using the YANG modeling language and in close collaboration with the NETMOD working group (see Section 2.4.2). The model specifies the necessary data for configuring and monitoring selection processes, caches, exporting processes, and collecting processes of IPFIX and PSAMP compliant monitoring devices.

At the time of this writing the NETMOD working group is developing core system and interface models in YANG.

The CAPWAP protocol exchanges Type Length Values (TLV). The base TLVs are specified in [RFC5415], while the TLVs for IEEE 802.11 are specified in [RFC5416]. CAPWAP Base MIB [RFC5833] specifies managed objects for modeling the CAPWAP Protocol and provides configuration and WTP status-monitoring aspects of CAPWAP, where CAPWAP Binding MIB [RFC5834] defines managed objects for modeling of CAPWAP protocol for IEEE 802.11 wireless binding.
Note: RFC 5833 and RFC 5834 have been published as Informational RFCs to provide the basis for future work on a SNMP management of the CAPWAP protocol.

4.2.3. Accounting Management

Accounting management collects usage information of network resources. Note that IETF does not define any mechanisms related to billing and charging. Many technology specific MIBs (link layer, network layer, transport layer or application layer) contain counters but are not primarily targeted for accounting, and therefore not included in this section.

[RFC4670] 'RADIUS Accounting Client MIB for IPv6' defines RADIUS Accounting Client MIB objects that support version-neutral IP addressing formats.

[RFC4671] 'RADIUS Accounting Server MIB for IPv6' defines RADIUS Accounting Server MIB objects that support version-neutral IP

addressing formats.

IPFIX/PSAMP Information Elements:

As expressed in Section 2.3, the IPFIX architecture [RFC5470] defines components involved in IP flow measurement and reporting of information on IP flows. As such, IPFIX records provide fine-grained measurement data for flexible and detailed usage reporting and enable usage-based accounting.

The IPFIX Information Elements (IE) have been initially defined in the IPFIX Information Model [RFC5102] and registered at the IANA [IANA-IPFIX]. The IPFIX IEs are composed of two types:

- o IEs related to identification of IP flows such as header information, derived packet properties, IGP and BGP next hop IP address, BGP AS, etc., and

- o IEs related to counter and timestamps, such as per-flow counters (e.g. octet count, packet count), flow start times, flow end times, and flow duration, etc.

The Information Elements specified in the IPFIX information model [RFC5102] are used by the PSAMP protocol where applicable. Packet Sampling (PSAMP) Parameters defined in the PSAMP protocol specification are registered at [IANA-PSAMP]. An additional set of PSAMP Information Elements for reporting packet information with the IPFIX/PSAMP protocol such as Sampling-related IEs are specified in the PSAMP Information Model [RFC5477]. These IEs fulfill the requirements on reporting of different sampling and filtering techniques specified in [RFC5475].

4.2.4. Performance Management

Performance management covers a set of functions that evaluate and report the performance of network elements and the network, with the goal to maintain the overall network performance at a defined level. Performance management functionality includes monitoring and measurement of network performance parameters, gathering statistical information, maintaining and examining activity logs. The data models below can be used for performance management tasks.

The RMON (Remote Network Monitoring) MIB [STD59][RFC2819] defines objects for collecting data related to network performance and traffic from remote monitoring devices. An organization may employ many remote monitoring probes, one per network segment, to monitor its network. These devices may be used by a network service provider to access a client network, often geographically remote. Most of the

objects in the RMON MIB module are suitable for the monitoring of any type of network, while some of them are specific to the monitoring of Ethernet networks.

RMON allows a probe to be configured to perform diagnostics and to collect network statistics continuously, even when communication with the management station may not be possible or efficient. The alarm group periodically takes statistical samples from variables in the probe and compares them to previously configured thresholds. If the monitored variable crosses a threshold, an event is generated.

[RFC3577] 'Introduction to the Remote Monitoring (RMON) Family of MIB Modules' describes the documents associated with the RMON framework and how they relate to each other.

The RMON-2 MIB [RFC4502] extends RMON by providing RMON analysis up to the application layer and defines performance data to monitor. The SMON MIB [RFC2613] extends RMON by providing RMON analysis for switched networks.

RMON MIB Extensions for High Capacity Alarms [RFC3434] describes managed objects for extending the alarm thresholding capabilities found in the RMON MIB and provides similar threshold monitoring of objects based on the Counter64 data type.

RMON MIB Extensions for High Capacity Networks [RFC3273] defines objects for managing RMON devices for use on high-speed networks.

RMON MIB Extensions for Interface Parameters Monitoring [RFC3144] describes an extension to the RMON MIB with a method of sorting the interfaces of a monitored device according to values of parameters specific to this interface.

[RFC4710] describes Real-Time Application Quality of Service Monitoring (RAQMON), which is part of the RMON protocol family. RAQMON supports end-to-end QoS monitoring for multiple concurrent applications and does not relate to a specific application transport. RAQMON is scalable and works well with encrypted payload and signaling. RAQMON uses TCP to transport RAQMON PDUs.

[RFC4711] proposes an extension to the Remote Monitoring MIB [STD59][RFC2819] and describes managed objects used for RAQMON. [RFC4712] specifies two transport mappings for the RAQMON information model using TCP as a native transport and SNMP to carry the RAQMON information from a RAQMON Data Source (RDS) to a RAQMON Report Collector (RRC).

Application Performance Measurement MIB [RFC3729] uses the

architecture created in the RMON MIB and defines objects by providing measurement and analysis of the application performance as experienced by end-users. [RFC3729] enables the measurement of the quality of service delivered to end-users by applications.

Transport Performance Metrics MIB [RFC4150] describes managed objects used for monitoring selectable performance metrics and statistics derived from the monitoring of network packets and sub-application level transactions. The metrics can be defined through reference to existing IETF, ITU, and other standards organizations' documents.

The IPPM working group has defined [RFC4148] "IP Performance Metrics (IPPM) Metrics Registry". Note that with the publication of [RFC6248], [RFC4148] and the corresponding IANA registry for IPPM metrics have been declared Obsolete and shouldn't be used.

The IPPM working group defined an Information Model and XML Data Model for Traceroute Measurements [RFC5388], which defines a common information model dividing the information elements into two semantically separated groups (configuration elements and results elements) with an additional element to relate configuration elements and results elements by means of a common unique identifier. Based on the information model, an XML data model is provided to store the results of traceroute measurements.

SIP Package for Voice Quality Reporting [RFC6035] defines a SIP event package that enables the collection and reporting of metrics that measure the quality for Voice over Internet Protocol (VoIP) sessions.

4.2.5. Security Management

The security management provides the set of functions to protect the network and system from unauthorized access and includes functions such as creating, deleting, and controlling security services and mechanisms; key management, reporting security-relevant events, and authorizing user access and privileges. Based on their support for authentication and authorization, RADIUS and DIAMETER are seen as security management protocols. The data models below can be used to utilize security management.

[RFC3414], part of [STD62], specifies the procedures for providing SNMPv3 message level security and includes a MIB module for remotely monitoring and managing the configuration parameters for the USM security model.

[RFC3415], part of [STD62], describes the procedures for controlling access to management information in the SNMPv3 architecture and includes a MIB module, which defines managed objects to access

portions of an SNMP engine's Local Configuration Datastore (LCD). As such, this MIB module enables remote management of the configuration parameters of the View-based Access Control Model.

NETCONF Access Control Model (NACM) [RFC6536] addresses the need for access control mechanisms for the operation and content layers of NETCONF, as defined in [RFC6241]. As such NACM proposes standard mechanisms to restrict NETCONF protocol access for particular users to a pre-configured subset of all available NETCONF protocol operations and content within a particular server.

There are numerous MIB modules defined for multiple purposes to use with RADIUS:

- [RFC4668] 'RADIUS Authentication Client MIB for IPv6' defines RADIUS Authentication Client MIB objects that support version-neutral IP addressing formats and defines a set of extensions for RADIUS authentication client functions.

- [RFC4669] 'RADIUS Authentication Server MIB for IPv6' defines RADIUS Authentication Server MIB objects that support version-neutral IP addressing formats and defines a set of extensions for RADIUS authentication server functions.

- [RFC4672] 'RADIUS Dynamic Authorization Client MIB' defines the MIB module for entities implementing the client side of the Dynamic Authorization Extensions to RADIUS [RFC5176].

- [RFC4673] 'RADIUS Dynamic Authorization Server MIB' defines the MIB module for entities implementing the server side of the Dynamic Authorization Extensions to RADIUS [RFC5176].

The MIB Module definitions in [RFC4668], [RFC4669], [RFC4672], [RFC4673] are intended to be used only for RADIUS over UDP and do not support RADIUS over TCP. There is also a recommendation that RADIUS clients and servers implementing RADIUS over TCP should not reuse earlier listed MIB modules to perform statistics counting for RADIUS over TCP connections.

Currently there are no standardized MIB modules for DIAMETER applications, which can be considered as a lack on the management side of DIAMETER nodes. There are ongoing efforts to produce standard MIBs for the 'Diameter Base Protocol' and the 'Diameter Credit-Control Application'.

5. IANA Considerations

This document does not introduce any new code-points or namespaces for registration with IANA.

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

6. Security Considerations

This document gives an overview of IETF network management standards and summarizes existing and ongoing development of IETF standards-track network management protocols and data models. As such it does not have any security implications in or of itself.

For each specific technology discussed in the document a summary of its security usage has been given in corresponding chapters. In a few cases, e.g. for SNMP, a detailed description of developed security mechanisms has been provided.

The attention of the reader is particularly drawn to the security discussion in following document sections:

- SNMP Security and Access Control Models in Section 2.1.4.1,
- User-Based Security Model (USM) in Section 2.1.4.2,
- View-Based Access Control Model (VACM) in Section 2.1.4.3,
- SNMP Transport Security Model in Section 2.1.5.1,
- Secure SYSLOG message delivery in Section 2.2,
- Use of secure NETCONF message transport and the NETCONF Access Control Model (NACM) in Section 2.4.1,
- Message authentication for Dynamic Host Configuration Protocol (DHCP) in Section 3.1.1,
- Security for Remote Authentication Dial In User Service (RADIUS) in conjunction with EAP and IEEE 802.1X authenticators in Section 3.5,
- Built in and transport security for Diameter Base Protocol (DIAMETER) in Section 3.6,
- Transport security for Control And Provisioning of Wireless Access Points (CAPWAP) in Section 3.7,

- Built in security for Access Node Control Protocol (ANCP) in Section 3.8,
- Security for Application Configuration Access Protocol (ACAP) in Section 3.9,
- Security for XML Configuration Access Protocol (XCAP) in Section 3.10, and
- Data models for the Security Management in Section 4.2.5.

The authors would like to refer also to detailed security consideration sections for specific management standards described in this document, which contain comprehensive discussion of security implications of the particular management protocols and mechanisms. Among others security consideration sections of following documents should be carefully read before implementing the technology.

- For SNMP security in general, subsequent security consideration sections in [STD62], which includes RFCs 3411-3418,
- Security consideration section in Section 8. of [BCP74] for the coexistence between SNMP v1, v2, and v3,
- Security considerations for the SNMP Transport Security Model in Section 8. of [RFC5591],
- Security considerations for the Secure Shell Transport Model for SNMP in Section 9 of [RFC5592],
- Security considerations for the TLS Transport Model for SNMP in Section 9. of [RFC6353],
- Security considerations for the TLS Transport Mapping for Syslog in Section 6 of [RFC5425],
- Security considerations for the IPFIX Protocol Specification in Section 11. of [RFC5101],
- Security considerations for the NETCONF protocol in Section 9. of [RFC6241] and the SSH transport in Section 6. of [RFC6242],
- Security considerations for the NETCONF Access Control Model (NACM) in Section 3.7. of [RFC6536],
- Security considerations for DHCPv4 and DHCPv6 in Section 7. of [RFC2131] and Section 23. of [RFC3315],

- Security considerations for RADIUS in Section 8. of [RFC2865],
- Security considerations for DIAMETER in Section 13. of [RFC3588],
- Security considerations for the CAPWAP protocol in Section 12. of [RFC5415],
- Security considerations for the ANCP protocol in Section 11. of [RFC6320], and
- Security considerations for the XCAP protocol in Section 14. of [RFC4825].

7. Contributors

Following persons made significant contributions to and reviewed this document:

- Ralph Droms (Cisco) - revised the section on IP address management and DHCP.
- Jouni Korhonen (Nokia Siemens Networks) - contributed the sections on RADIUS and DIAMETER.
- Al Morton (AT&T) - contributed to the section on IP Performance Metrics.
- Juergen Quittek (NEC) - contributed the section on IPFIX/PSAMP.
- Juergen Schoenwaelder (Jacobs University Bremen) - contributed the sections on IETF Network Management Data Models and YANG.

8. Acknowledgements

The editor would like to thank to Fred Baker, Alex Clemm, Miguel A. Garcia, Simon Leinen, Christopher Liljenstolpe, Tom Petch, Randy Presuhn, Dan Romascanu, Juergen Schoenwaelder, Tina Tsou, and Henk Uijterwaal, for their valuable suggestions, comments in the OPSAWG sessions and mailing list.

The editor would like to especially thank Dave Harrington, who created the document "Survey of IETF Network Management Standards" a few years ago, which has been used as a starting point and enhanced with a special focus on the description of the IETF network management standards and management data models.

9. Informative References

[3GPPEPC] 3GPP, "Access to the 3GPP Evolved Packet Core (EPC) via non-3GPP access networks", December 2010, <http://www.3gpp.org/ftp/Specs/html-info/24302.htm>.

[3GPPIMS] 3GPP, "Release 10, IP Multimedia Subsystem (IMS); Stage 2", September 2010, <http://www.3gpp.org/ftp/Specs/html-info/23228.htm>.

[BCP170] Clark, A. and B. Claise, "Guidelines for Considering New Performance Metric Development", October 2011.

[BCP27] D. O'Dell, M., "Advancement of MIB specifications on the IETF Standards Track", October 1998.

[BCP74] Frye, R., "Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework", August 2003.

[DMTF-CIM] DMTF, "Common Information Model Schema, Version 2.27.0", November 2010, <http://www.dmtf.org/standards/cim>.

[FCAPS] International Telecommunication Union, "X.700: Management Framework For Open Systems Interconnection

(OSI) For CCITT Applications", September 1992, <http://www.itu.int/rec/T-REC-X.700-199209-I/en>.

[I-D.ietf-dime-rfc3588bis] Fajardo, V., Arkko, J., Loughney, J., and G. Zorn, "Diameter Base Protocol", draft-ietf-dime-rfc3588bis-31 (work in progress), March 2012.

[I-D.ietf-ipfix-configuration-model] Muenz, G., Claise, B., and P. Aitken, "Configuration Data Model for IPFIX and PSAMP", draft-ietf-ipfix-configuration-model-10 (work in progress), July 2011.

[I-D.ietf-ipfix-export-per-sctp-stream] Claise, B., Aitken, P., Johnson, A., and G. Muenz, "IPFIX Export per SCTP Stream", draft-ietf-ipfix-export-per-sctp-stream-08 (work in progress), June 2010.

[I-D.ietf-ipfix-psamp-mib] Dietz, T., Claise, B., and J. Quittek, "Definitions of Managed Objects for Packet Sampling", draft-ietf-ipfix-psamp-mib-04 (work in progress), October 2011.

[I-D.ietf-mpls-tp-mib-management-overview] King, D. and V. Mahalingam, "Multiprotocol Label Switching Transport Profile (MPLS-TP) MIB-based Management Overview", draft-ietf-mpls-tp-mib-management-overview-07 (work in

progress), March 2012.

[I-D.ietf-mpls-tp-oam-analysis] Sprecher, N. and L. Fang, "An Overview of the OAM Tool Set for MPLS based Transport Networks", draft-ietf-mpls-tp-oam-analysis-08 (work in progress), March 2012.

[I-D.ietf-opsawg-oam-overview] Mizrahi, T., Sprecher, N., Bellagamba, E., and Y. Weingarten, "An Overview of Operations, Administration, and Maintenance (OAM) Mechanisms", draft-ietf-opsawg-oam-overview-06 (work in progress), March 2012.

[I-D.weil-shared-transition-space-request] Weil, J., Kuarsingh, V., Donley, C., Liljenstolpe, C., and M. Azinger, "IANA Reserved IPv4 Prefix for Shared Address Space", draft-weil-shared-transition-space-request-15 (work in progress), February 2012.

[IANA-AAA] Internet Assigned Numbers Authority, "IANA AAA Parameters", June 2011, <http://www.iana.org/assignments/aaa-parameters/aaa-parameters.xml>.

[IANA-IPFIX] Internet Assigned Numbers Authority, "IANA IPFIX Information Elements", February 2011, <http://www.iana.org/assignments/ipfix/ipfix.xml>.

[IANA-PROT] Internet Assigned Numbers Authority, "IANA Protocol

Registries", October 2010, <http://www.iana.org/protocols/>.

[IANA-PSAMP] Internet Assigned Numbers Authority, "IANA PSAMP Parameters", April 2009, <http://www.iana.org/assignments/psamp-parameters/psamp-parameters.xml>.

[IETF-WGS] IETF, "IETF Working Groups", <http://datatracker.ietf.org/wg/>.

[ITU-M3100] International Telecommunication Union, "M.3100: Generic network information model", January 2006, <http://www.itu.int/rec/T-REC-M.3100-200504-I>.

[ITU-X680] International Telecommunication Union, "X.680: Abstract Syntax Notation One (ASN.1): Specification of basic notation", July 2002, <http://www.itu.int/ITU-T/studygroups/com17/languages/X.680-0207.pdf>.

[ITU-X733] International Telecommunication Union, "X.733: Systems Management: Alarm Reporting Function", October 1992, <http://www.itu.int/rec/T-REC-X.733-199202-I/en>.

[RELAX-NG] OASIS, "RELAX NG Specification, Committee Specification 3 December

2001", December 2001, <http://www.oasis-open.org/committees/relax-ng/spec-20011203.html>.

[RFC0951] Croft, B. and J. Gilmore, "Bootstrap Protocol", RFC 951, September 1985.

[RFC1021] Partridge, C. and G. Trewitt, "High-level Entity Management System (HEMS)", RFC 1021, October 1987.

[RFC1155] Rose, M. and K. McCloghrie, "Structure and identification of management information for TCP/IP-based internets", STD 16, RFC 1155, May 1990.

[RFC1157] Case, J., Fedor, M., Schoffstall, M., and J. Davin, "Simple Network Management Protocol (SNMP)", STD 15, RFC 1157, May 1990.

[RFC1212] Rose, M. and K. McCloghrie, "Concise MIB definitions", STD 16, RFC 1212, March 1991.

[RFC1215] Rose, M., "Convention for defining traps for use with the SNMP", RFC 1215, March 1991.

[RFC1321] Rivest, R., "The MD5 Message-Digest Algorithm", RFC 1321, April 1992.

[RFC1470] Enger, R. and J. Reynolds, "FYI on a Network Management Tool

Catalog: Tools for Monitoring and Debugging TCP/IP Internets and Interconnected Devices", RFC 1470, June 1993.

[RFC1901] Case, J., McCloghrie, K., McCloghrie, K., Rose, M., and S. Waldbusser, "Introduction to Community-based SNMPv2", RFC 1901, January 1996.

[RFC2026] Bradner, S., "The Internet Standards Process -- Revision 3", BCP 9, RFC 2026, October 1996.

[RFC2127] Roeck, G., "ISDN Management Information Base using SMIv2", RFC 2127, March 1997.

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, March 1997.

[RFC2195] Klensin, J., Catoe, R., and P. Krumviede, "IMAP/POP AUTHorize Extension for Simple Challenge/Response", RFC 2195, September 1997.

[RFC2244] Newman, C. and J. Myers, "ACAP -- Application Configuration Access Protocol", RFC 2244, November 1997.

[RFC2287] Krupczak, C. and J. Saperia, "Definitions of System-Level Managed Objects for Applications", RFC 2287, February 1998.

[RFC2330] Paxson, V., Almes, G., Mahdavi, J., and M. Mathis, "Framework for IP Performance Metrics", RFC 2330, May 1998.

[RFC2458] Lu, H., Krishnaswamy, M., Conroy, L., Bellovin, S., Burg, F., DeSimone, A., Tewani, K., Davidson, P., Schulzrinne, H., and K. Vishwanathan, "Toward the PSTN/Internet Inter-Networking --Pre-PINT Implementations", RFC 2458, November 1998.

[RFC2515] Tesink, K., "Definitions of Managed Objects for ATM Management", RFC 2515, February 1999.

[RFC2564] Kalbfleisch, C., Krupczak, C., Presuhn, R., and J. Saperia, "Application Management MIB", RFC 2564, May 1999.

[RFC2578] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, April 1999.

[RFC2579] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Textual Conventions for SMIv2", STD 58, RFC 2579, April 1999.

[RFC2580] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Conformance Statements for SMIv2", STD 58, RFC 2580, April 1999.

[RFC2610] Perkins, C. and E. Guttman, "DHCP Options for Service Location Protocol", RFC 2610, June 1999.

[RFC2613] Waterman, R., Lahaye, B., Romascanu, D., and S. Waldbusser, "Remote Network Monitoring MIB Extensions for Switched Networks Version 1.0", RFC 2613, June 1999.

[RFC2617] Franks, J., Hallam-Baker, P., Hostetler, J., Lawrence, S., Leach, P., Luotonen, A., and L. Stewart, "HTTP Authentication: Basic and Digest Access Authentication", RFC 2617, June 1999.

[RFC2678] Mahdavi, J. and V. Paxson, "IPPM Metrics for Measuring Connectivity", RFC 2678, September 1999.

[RFC2679] Almes, G., Kalidindi, S., and M. Zekauskas, "A One-way Delay Metric for IPPM", RFC 2679, September 1999.

[RFC2680] Almes, G., Kalidindi, S., and M. Zekauskas, "A One-way Packet Loss Metric for IPPM", RFC 2680, September 1999.

[RFC2681] Almes, G., Kalidindi, S., and M. Zekauskas, "A Round-trip Delay Metric for IPPM", RFC 2681, September 1999.

[RFC2748] Durham, D., Boyle, J.,

Cohen, R., Herzog, S., Rajan, R., and A. Sastry, "The COPS (Common Open Policy Service) Protocol", RFC 2748, January 2000.

[RFC2753] Yavatkar, R., Pendarakis, D., and R. Guerin, "A Framework for Policy-based Admission Control", RFC 2753, January 2000.

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, May 2000.

[RFC2819] Waldbusser, S., "Remote Network Monitoring Management Information Base", STD 59, RFC 2819, May 2000.

[RFC2863] McCloghrie, K. and F. Kastenholz, "The Interfaces Group MIB", RFC 2863, June 2000.

[RFC2865] Rigney, C., Willens, S., Rubens, A., and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, June 2000.

[RFC2866] Rigney, C., "RADIUS Accounting", RFC 2866, June 2000.

[RFC2867] Zorn, G., Aboba, B., and D. Mitton, "RADIUS Accounting Modifications for Tunnel Protocol Support", RFC 2867, June 2000.

[RFC2868] Zorn, G., Leifer, D., Rubens, A., Shriver, J., Holdrege, M., and I.

Goyret, "RADIUS Attributes for Tunnel Protocol Support", RFC 2868, June 2000.

[RFC2869] Rigney, C., Willats, W., and P. Calhoun, "RADIUS Extensions", RFC 2869, June 2000.

[RFC2981] Kavasseri, R., "Event MIB", RFC 2981, October 2000.

[RFC2982] Kavasseri, R., "Distributed Management Expression MIB", RFC 2982, October 2000.

[RFC3014] Kavasseri, R., "Notification Log MIB", RFC 3014, November 2000.

[RFC3046] Patrick, M., "DHCP Relay Agent Information Option", RFC 3046, January 2001.

[RFC3084] Chan, K., Seligson, J., Durham, D., Gai, S., McCloghrie, K., Herzog, S., Reichmeyer, F., Yavatkar, R., and A. Smith, "COPS Usage for Policy Provisioning (COPS-PR)", RFC 3084, March 2001.

[RFC3144] Romascanu, D., "Remote Monitoring MIB Extensions for Interface Parameters Monitoring", RFC 3144, August 2001.

[RFC3159] McCloghrie, K., Fine, M., Seligson, J., Chan, K., Hahn, S., Sahita, R., Smith, A., and F.

Reichmeyer, "Structure of Policy Provisioning Information (SPPI)", RFC 3159, August 2001.

[RFC3162] Aboba, B., Zorn, G., and D. Mitton, "RADIUS and IPv6", RFC 3162, August 2001.

[RFC3164] Lonvick, C., "The BSD Syslog Protocol", RFC 3164, August 2001.

[RFC3165] Levi, D. and J. Schoenwaelder, "Definitions of Managed Objects for the Delegation of Management Scripts", RFC 3165, August 2001.

[RFC3195] New, D. and M. Rose, "Reliable Delivery for syslog", RFC 3195, November 2001.

[RFC3273] Waldbusser, S., "Remote Network Monitoring Management Information Base for High Capacity Networks", RFC 3273, July 2002.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

[RFC3319] Schulzrinne, H. and B. Volz, "Dynamic Host Configuration Protocol (DHCPv6) Options for Session Initiation Protocol (SIP) Servers",

RFC 3319, July 2003.

[RFC3393] Demichelis, C. and P. Chimento, "IP Packet Delay Variation Metric for IP Performance Metrics (IPPM)", RFC 3393, November 2002.

[RFC3410] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", RFC 3410, December 2002.

[RFC3411] Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks", STD 62, RFC 3411, December 2002.

[RFC3413] Levi, D., Meyer, P., and B. Stewart, "Simple Network Management Protocol (SNMP) Applications", STD 62, RFC 3413, December 2002.

[RFC3414] Blumenthal, U. and B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", STD 62, RFC 3414, December 2002.

[RFC3415] Wijnen, B., Presuhn, R., and K. McCloghrie, "View-based Access Control Model (VACM) for the Simple Network Management

Protocol (SNMP)", STD 62, RFC 3415, December 2002.

[RFC3417] Presuhn, R., "Transport Mappings for the Simple Network Management Protocol (SNMP)", STD 62, RFC 3417, December 2002.

[RFC3418] Presuhn, R., "Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)", STD 62, RFC 3418, December 2002.

[RFC3430] Schoenwaelder, J., "Simple Network Management Protocol Over Transmission Control Protocol Transport Mapping", RFC 3430, December 2002.

[RFC3432] Raisanen, V., Grotefeld, G., and A. Morton, "Network performance measurement with periodic streams", RFC 3432, November 2002.

[RFC3433] Bierman, A., Romascanu, D., and K. Norseth, "Entity Sensor Management Information Base", RFC 3433, December 2002.

[RFC3434] Bierman, A. and K. McCloghrie, "Remote Monitoring MIB Extensions for High Capacity Alarms", RFC 3434, December 2002.

[RFC3444] Pras, A. and J. Schoenwaelder, "On the Difference between Information Models and

Data Models", RFC 3444, January 2003.

[RFC3460] Moore, B., "Policy Core Information Model (PCIM) Extensions", RFC 3460, January 2003.

[RFC3535] Schoenwaelder, J., "Overview of the 2002 IAB Network Management Workshop", RFC 3535, May 2003.

[RFC3574] Soininen, J., "Transition Scenarios for 3GPP Networks", RFC 3574, August 2003.

[RFC3577] Waldbusser, S., Cole, R., Kalbfleisch, C., and D. Romascanu, "Introduction to the Remote Monitoring (RMON) Family of MIB Modules", RFC 3577, August 2003.

[RFC3579] Aboba, B. and P. Calhoun, "RADIUS (Remote Authentication Dial In User Service) Support For Extensible Authentication Protocol (EAP)", RFC 3579, September 2003.

[RFC3580] Congdon, P., Aboba, B., Smith, A., Zorn, G., and J. Roese, "IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines", RFC 3580, September 2003.

[RFC3588] Calhoun, P., Loughney, J., Guttman, E., Zorn, G., and J. Arkko, "Diameter Base Protocol",

RFC 3588, September 2003.

[RFC3589] Loughney, J., "Diameter Command Codes for Third Generation Partnership Project (3GPP) Release 5", RFC 3589, September 2003.

[RFC3606] Ly, F., Noto, M., Smith, A., Spiegel, E., and K. Tesink, "Definitions of Supplemental Managed Objects for ATM Interface", RFC 3606, November 2003.

[RFC3633] Troan, O. and R. Droms, "IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6", RFC 3633, December 2003.

[RFC3646] Droms, R., "DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3646, December 2003.

[RFC3729] Waldbusser, S., "Application Performance Measurement MIB", RFC 3729, March 2004.

[RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., and H. Levkowetz, "Extensible Authentication Protocol (EAP)", RFC 3748, June 2004.

[RFC3758] Stewart, R., Ramalho, M., Xie, Q., Tuexen, M., and P. Conrad, "Stream Control Transmission

Protocol (SCTP) Partial Reliability Extension", RFC 3758, May 2004.

[RFC3868] Loughney, J., Sidebottom, G., Coene, L., Verwimp, G., Keller, J., and B. Bidulock, "Signalling Connection Control Part User Adaptation Layer (SUA)", RFC 3868, October 2004.

[RFC3873] Pastor, J. and M. Belinchon, "Stream Control Transmission Protocol (SCTP) Management Information Base (MIB)", RFC 3873, September 2004.

[RFC3877] Chisholm, S. and D. Romascanu, "Alarm Management Information Base (MIB)", RFC 3877, September 2004.

[RFC3878] Lam, H., Huynh, A., and D. Perkins, "Alarm Reporting Control Management Information Base (MIB)", RFC 3878, September 2004.

[RFC3917] Quittek, J., Zseby, T., Claise, B., and S. Zander, "Requirements for IP Flow Information Export (IPFIX)", RFC 3917, October 2004.

[RFC3954] Claise, B., "Cisco Systems NetFlow Services Export Version 9", RFC 3954, October 2004.

[RFC4004] Calhoun, P., Johansson, T., Perkins, C., Hiller,

T., and P. McCann, "Diameter Mobile IPv4 Application", RFC 4004, August 2005.

[RFC4005] Calhoun, P., Zorn, G., Spence, D., and D. Mitton, "Diameter Network Access Server Application", RFC 4005, August 2005.

[RFC4006] Hakala, H., Mattila, L., Koskinen, J-P., Stura, M., and J. Loughney, "Diameter Credit-Control Application", RFC 4006, August 2005.

[RFC4022] Raghunarayan, R., "Management Information Base for the Transmission Control Protocol (TCP)", RFC 4022, March 2005.

[RFC4029] Lind, M., Ksinant, V., Park, S., Baudot, A., and P. Savola, "Scenarios and Analysis for Introducing IPv6 into ISP Networks", RFC 4029, March 2005.

[RFC4038] Shin, M-K., Hong, Y-G., Hagino, J., Savola, P., and E. Castro, "Application Aspects of IPv6 Transition", RFC 4038, March 2005.

[RFC4057] Bound, J., "IPv6 Enterprise Network Scenarios", RFC 4057, June 2005.

[RFC4072] Eronen, P., Hiller, T., and G. Zorn, "Diameter Extensible Authentication Protocol (EAP)

Application", RFC 4072, August 2005.

[RFC4113] Fenner, B. and J. Flick, "Management Information Base for the User Datagram Protocol (UDP)", RFC 4113, June 2005.

[RFC4118] Yang, L., Zerfos, P., and E. Sadot, "Architecture Taxonomy for Control and Provisioning of Wireless Access Points (CAPWAP)", RFC 4118, June 2005.

[RFC4133] Bierman, A. and K. McCloghrie, "Entity MIB (Version 3)", RFC 4133, August 2005.

[RFC4148] Stephan, E., "IP Performance Metrics (IPPM) Metrics Registry", BCP 108, RFC 4148, August 2005.

[RFC4150] Dietz, R. and R. Cole, "Transport Performance Metrics MIB", RFC 4150, August 2005.

[RFC4188] Norseth, K. and E. Bell, "Definitions of Managed Objects for Bridges", RFC 4188, September 2005.

[RFC4213] Nordmark, E. and R. Gilligan, "Basic Transition Mechanisms for IPv6 Hosts and Routers", RFC 4213, October 2005.

[RFC4215] Wiljakka, J., "Analysis on IPv6 Transition in Third Generation Partnership Project (3GPP) Networks",

RFC 4215, October 2005.

[RFC4221] Nadeau, T., Srinivasan, C., and A. Farrel, "Multiprotocol Label Switching (MPLS) Management Overview", RFC 4221, November 2005.

[RFC4268] Chisholm, S. and D. Perkins, "Entity State MIB", RFC 4268, November 2005.

[RFC4273] Haas, J. and S. Hares, "Definitions of Managed Objects for BGP-4", RFC 4273, January 2006.

[RFC4280] Chowdhury, K., Yegani, P., and L. Madour, "Dynamic Host Configuration Protocol (DHCP) Options for Broadcast and Multicast Control Servers", RFC 4280, November 2005.

[RFC4285] Patel, A., Leung, K., Khalil, M., Akhtar, H., and K. Chowdhury, "Authentication Protocol for Mobile IPv6", RFC 4285, January 2006.

[RFC4292] Haberman, B., "IP Forwarding Table MIB", RFC 4292, April 2006.

[RFC4293] Routhier, S., "Management Information Base for the Internet Protocol (IP)", RFC 4293, April 2006.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301,

December 2005.

[RFC4318] Levi, D. and D. Harrington, "Definitions of Managed Objects for Bridges with Rapid Spanning Tree Protocol", RFC 4318, December 2005.

[RFC4363] Levi, D. and D. Harrington, "Definitions of Managed Objects for Bridges with Traffic Classes, Multicast Filtering, and Virtual LAN Extensions", RFC 4363, January 2006.

[RFC4422] Melnikov, A. and K. Zeilenga, "Simple Authentication and Security Layer (SASL)", RFC 4422, June 2006.

[RFC4444] Parker, J., "Management Information Base for Intermediate System to Intermediate System (IS-IS)", RFC 4444, April 2006.

[RFC4502] Waldbusser, S., "Remote Network Monitoring Management Information Base Version 2", RFC 4502, May 2006.

[RFC4546] Raftus, D. and E. Cardona, "Radio Frequency (RF) Interface Management Information Base for Data over Cable Service Interface Specifications (DOCSIS) 2.0 Compliant RF Interfaces", RFC 4546, June 2006.

[RFC4560] Quittek, J. and K. White,

"Definitions of Managed Objects for Remote Ping, Traceroute, and Lookup Operations", RFC 4560, June 2006.

[RFC4564] Govindan, S., Cheng, H., Yao, ZH., Zhou, WH., and L. Yang, "Objectives for Control and Provisioning of Wireless Access Points (CAPWAP)", RFC 4564, July 2006.

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

[RFC4663] Harrington, D., "Transferring MIB Work from IETF Bridge MIB WG to IEEE 802.1 WG", RFC 4663, September 2006.

[RFC4668] Nelson, D., "RADIUS Authentication Client MIB for IPv6", RFC 4668, August 2006.

[RFC4669] Nelson, D., "RADIUS Authentication Server MIB for IPv6", RFC 4669, August 2006.

[RFC4670] Nelson, D., "RADIUS Accounting Client MIB for IPv6", RFC 4670, August 2006.

[RFC4671] Nelson, D., "RADIUS Accounting Server MIB for IPv6", RFC 4671, August 2006.

[RFC4672] De Cnodder, S., Jonnala,

N., and M. Chiba, "RADIUS Dynamic Authorization Client MIB", RFC 4672, September 2006.

[RFC4673] De Cnodder, S., Jonnala, N., and M. Chiba, "RADIUS Dynamic Authorization Server MIB", RFC 4673, September 2006.

[RFC4675] Congdon, P., Sanchez, M., and B. Aboba, "RADIUS Attributes for Virtual LAN and Priority Support", RFC 4675, September 2006.

[RFC4706] Morgenstern, M., Dodge, M., Baillie, S., and U. Bonollo, "Definitions of Managed Objects for Asymmetric Digital Subscriber Line 2 (ADSL2)", RFC 4706, November 2006.

[RFC4710] Siddiqui, A., Romascanu, D., and E. Golovinsky, "Real-time Application Quality-of-Service Monitoring (RAQMON) Framework", RFC 4710, October 2006.

[RFC4711] Siddiqui, A., Romascanu, D., and E. Golovinsky, "Real-time Application Quality-of-Service Monitoring (RAQMON) MIB", RFC 4711, October 2006.

[RFC4712] Siddiqui, A., Romascanu, D., Golovinsky, E., Rahman, M., and Y. Kim, "Transport Mappings for Real-time Application Quality-of-Service

Monitoring (RAQMON) Protocol Data Unit (PDU)", RFC 4712, October 2006.

[RFC4737] Morton, A., Ciavattone, L., Ramachandran, G., Shalunov, S., and J. Perser, "Packet Reordering Metrics", RFC 4737, November 2006.

[RFC4740] Garcia-Martin, M., Belinchon, M., Pallares-Lopez, M., Canales-Valenzuela, C., and K. Tammi, "Diameter Session Initiation Protocol (SIP) Application", RFC 4740, November 2006.

[RFC4743] Goddard, T., "Using NETCONF over the Simple Object Access Protocol (SOAP)", RFC 4743, December 2006.

[RFC4744] Lear, E. and K. Crozier, "Using the NETCONF Protocol over the Blocks Extensible Exchange Protocol (BEEP)", RFC 4744, December 2006.

[RFC4750] Joyal, D., Galecki, P., Giacalone, S., Coltun, R., and F. Baker, "OSPF Version 2 Management Information Base", RFC 4750, December 2006.

[RFC4780] Lingle, K., Mule, J-F., Maeng, J., and D. Walker, "Management Information Base for the Session Initiation Protocol (SIP)", RFC 4780, April 2007.

[RFC4789] Schoenwaelder, J. and T. Jeffree, "Simple Network Management Protocol (SNMP) over IEEE 802 Networks", RFC 4789, November 2006.

[RFC4803] Nadeau, T. and A. Farrel, "Generalized Multiprotocol Label Switching (GMPLS) Label Switching Router (LSR) Management Information Base", RFC 4803, February 2007.

[RFC4818] Salowey, J. and R. Droms, "RADIUS Delegated-IPv6-Prefix Attribute", RFC 4818, April 2007.

[RFC4825] Rosenberg, J., "The Extensible Markup Language (XML) Configuration Access Protocol (XCAP)", RFC 4825, May 2007.

[RFC4826] Rosenberg, J., "Extensible Markup Language (XML) Formats for Representing Resource Lists", RFC 4826, May 2007.

[RFC4827] Isomaki, M. and E. Leppanen, "An Extensible Markup Language (XML) Configuration Access Protocol (XCAP) Usage for Manipulating Presence Document Contents", RFC 4827, May 2007.

[RFC4898] Mathis, M., Heffner, J., and R. Raghunarayan, "TCP Extended Statistics MIB", RFC 4898, May 2007.

[RFC4960] Stewart, R., "Stream Control Transmission Protocol", RFC 4960, September 2007.

[RFC5060] Sivaramu, R., Lingard, J., McWalter, D., Joshi, B., and A. Kessler, "Protocol Independent Multicast MIB", RFC 5060, January 2008.

[RFC5080] Nelson, D. and A. DeKok, "Common Remote Authentication Dial In User Service (RADIUS) Implementation Issues and Suggested Fixes", RFC 5080, December 2007.

[RFC5085] Nadeau, T. and C. Pignataro, "Pseudowire Virtual Circuit Connectivity Verification (VCCV): A Control Channel for Pseudowires", RFC 5085, December 2007.

[RFC5090] Sterman, B., Sadolevsky, D., Schwartz, D., Williams, D., and W. Beck, "RADIUS Extension for Digest Authentication", RFC 5090, February 2008.

[RFC5101] Claise, B., "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information", RFC 5101, January 2008.

[RFC5102] Quittek, J., Bryant, S., Claise, B., Aitken, P., and J. Meyer, "Information Model for IP

Flow Information Export", RFC 5102, January 2008.

[RFC5103] Trammell, B. and E. Boschi, "Bidirectional Flow Export Using IP Flow Information Export (IPFIX)", RFC 5103, January 2008.

[RFC5176] Chiba, M., Dommety, G., Eklund, M., Mitton, D., and B. Aboba, "Dynamic Authorization Extensions to Remote Authentication Dial In User Service (RADIUS)", RFC 5176, January 2008.

[RFC5181] Shin, M-K., Han, Y-H., Kim, S-E., and D. Premec, "IPv6 Deployment Scenarios in 802.16 Networks", RFC 5181, May 2008.

[RFC5224] Brenner, M., "Diameter Policy Processing Application", RFC 5224, March 2008.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

[RFC5277] Chisholm, S. and H. Trevino, "NETCONF Event Notifications", RFC 5277, July 2008.

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

[RFC5388] Niccolini, S., Tartarelli, S., Quittek, J., Dietz, T., and M. Swany, "Information Model and XML Data Model for Traceroute Measurements", RFC 5388, December 2008.

[RFC5415] Calhoun, P., Montemurro, M., and D. Stanley, "Control And Provisioning of Wireless Access Points (CAPWAP) Protocol Specification", RFC 5415, March 2009.

[RFC5416] Calhoun, P., Montemurro, M., and D. Stanley, "Control and Provisioning of Wireless Access Points (CAPWAP) Protocol Binding for IEEE 802.11", RFC 5416, March 2009.

[RFC5424] Gerhards, R., "The Syslog Protocol", RFC 5424, March 2009.

[RFC5425] Miao, F., Ma, Y., and J. Salowey, "Transport Layer Security (TLS) Transport Mapping for Syslog", RFC 5425, March 2009.

[RFC5426] Okmianski, A., "Transmission of Syslog Messages over UDP", RFC 5426, March 2009.

[RFC5427] Keeni, G., "Textual Conventions for Syslog Management", RFC 5427, March 2009.

[RFC5431] Sun, D., "Diameter ITU-T Rw Policy Enforcement Interface Application", RFC 5431, March 2009.

[RFC5447] Korhonen, J., Bournelle, J., Tschofenig, H., Perkins, C., and K. Chowdhury, "Diameter Mobile IPv6: Support for Network Access Server to Diameter Server Interaction", RFC 5447, February 2009.

[RFC5470] Sadasivan, G., Brownlee, N., Claise, B., and J. Quittek, "Architecture for IP Flow Information Export", RFC 5470, March 2009.

[RFC5472] Zseby, T., Boschi, E., Brownlee, N., and B. Claise, "IP Flow Information Export (IPFIX) Applicability", RFC 5472, March 2009.

[RFC5473] Boschi, E., Mark, L., and B. Claise, "Reducing Redundancy in IP Flow Information Export (IPFIX) and Packet Sampling (PSAMP) Reports", RFC 5473, March 2009.

[RFC5474] Duffield, N., Chiou, D., Claise, B., Greenberg, A., Grossglauser, M., and J. Rexford, "A Framework for Packet Selection and Reporting", RFC 5474, March 2009.

[RFC5475] Zseby, T., Molina, M., Duffield, N., Niccolini, S., and F. Raspall, "Sampling and Filtering Techniques for IP Packet Selection", RFC 5475, March 2009.

[RFC5476] Claise, B., Johnson, A., and J. Quittek, "Packet Sampling (PSAMP) Protocol Specifications", RFC 5476, March 2009.

[RFC5477] Dietz, T., Claise, B., Aitken, P., Dressler, F., and G. Carle, "Information Model for Packet Sampling Exports", RFC 5477, March 2009.

[RFC5516] Jones, M. and L. Morand, "Diameter Command Code Registration for the Third Generation Partnership Project (3GPP) Evolved Packet System (EPS)", RFC 5516, April 2009.

[RFC5539] Badra, M., "NETCONF over Transport Layer Security (TLS)", RFC 5539, May 2009.

[RFC5560] Uijterwaal, H., "A One-Way Packet Duplication Metric", RFC 5560, May 2009.

[RFC5580] Tschofenig, H., Adrangi, F., Jones, M., Lior, A., and B. Aboba, "Carrying Location Objects in RADIUS and Diameter", RFC 5580, August 2009.

[RFC5590] Harrington, D. and J. Schoenwaelder, "Transport Subsystem for the Simple Network Management Protocol (SNMP)", RFC 5590, June 2009.

[RFC5591] Harrington, D. and W. Hardaker, "Transport

Security Model for the Simple Network Management Protocol (SNMP)", RFC 5591, June 2009.

[RFC5592] Harrington, D., Salowey, J., and W. Hardaker, "Secure Shell Transport Model for the Simple Network Management Protocol (SNMP)", RFC 5592, June 2009.

[RFC5607] Nelson, D. and G. Weber, "Remote Authentication Dial-In User Service (RADIUS) Authorization for Network Access Server (NAS) Management", RFC 5607, July 2009.

[RFC5608] Narayan, K. and D. Nelson, "Remote Authentication Dial-In User Service (RADIUS) Usage for Simple Network Management Protocol (SNMP) Transport Models", RFC 5608, August 2009.

[RFC5610] Boschi, E., Trammell, B., Mark, L., and T. Zseby, "Exporting Type Information for IP Flow Information Export (IPFIX) Information Elements", RFC 5610, July 2009.

[RFC5650] Morgenstern, M., Baillie, S., and U. Bonollo, "Definitions of Managed Objects for Very High Speed Digital Subscriber Line 2 (VDSL2)", RFC 5650, September 2009.

[RFC5655] Trammell, B., Boschi, E.,

Mark, L., Zseby, T., and A. Wagner, "Specification of the IP Flow Information Export (IPFIX) File Format", RFC 5655, October 2009.

[RFC5674] Chisholm, S. and R. Gerhards, "Alarms in Syslog", RFC 5674, October 2009.

[RFC5675] Marinov, V. and J. Schoenwaelder, "Mapping Simple Network Management Protocol (SNMP) Notifications to SYSLOG Messages", RFC 5675, October 2009.

[RFC5676] Schoenwaelder, J., Clemm, A., and A. Karmakar, "Definitions of Managed Objects for Mapping SYSLOG Messages to Simple Network Management Protocol (SNMP) Notifications", RFC 5676, October 2009.

[RFC5706] Harrington, D., "Guidelines for Considering Operations and Management of New Protocols and Protocol Extensions", RFC 5706, November 2009.

[RFC5713] Moustafa, H., Tschofenig, H., and S. De Cnodder, "Security Threats and Security Requirements for the Access Node Control Protocol (ANCP)", RFC 5713, January 2010.

[RFC5717] Lengyel, B. and M. Bjorklund, "Partial Lock

Remote Procedure Call (RPC) for NETCONF", RFC 5717, December 2009.

[RFC5719] Romascanu, D. and H. Tschofenig, "Updated IANA Considerations for Diameter Command Code Allocations", RFC 5719, January 2010.

[RFC5729] Korhonen, J., Jones, M., Morand, L., and T. Tsou, "Clarifications on the Routing of Diameter Requests Based on the Username and the Realm", RFC 5729, December 2009.

[RFC5777] Korhonen, J., Tschofenig, H., Arumaithurai, M., Jones, M., and A. Lior, "Traffic Classification and Quality of Service (QoS) Attributes for Diameter", RFC 5777, February 2010.

[RFC5778] Korhonen, J., Tschofenig, H., Bournelle, J., Giaretta, G., and M. Nakhjiri, "Diameter Mobile IPv6: Support for Home Agent to Diameter Server Interaction", RFC 5778, February 2010.

[RFC5779] Korhonen, J., Bournelle, J., Chowdhury, K., Muhanna, A., and U. Meyer, "Diameter Proxy Mobile IPv6: Mobile Access Gateway and Local Mobility Anchor Interaction with Diameter Server", RFC 5779, February 2010.

[RFC5815] Dietz, T., Kobayashi, A., Claise, B., and G. Muenz, "Definitions of Managed Objects for IP Flow Information Export", RFC 5815, April 2010.

[RFC5833] Shi, Y., Perkins, D., Elliott, C., and Y. Zhang, "Control and Provisioning of Wireless Access Points (CAPWAP) Protocol Base MIB", RFC 5833, May 2010.

[RFC5834] Shi, Y., Perkins, D., Elliott, C., and Y. Zhang, "Control and Provisioning of Wireless Access Points (CAPWAP) Protocol Binding MIB for IEEE 802.11", RFC 5834, May 2010.

[RFC5835] Morton, A. and S. Van den Berghe, "Framework for Metric Composition", RFC 5835, April 2010.

[RFC5848] Kelsey, J., Callas, J., and A. Clemm, "Signed Syslog Messages", RFC 5848, May 2010.

[RFC5851] Ooghe, S., Voigt, N., Platnic, M., Haag, T., and S. Wadhwa, "Framework and Requirements for an Access Node Control Mechanism in Broadband Multi-Service Networks", RFC 5851, May 2010.

[RFC5866] Sun, D., McCann, P., Tschofenig, H., Tsou, T., Doria, A., and G. Zorn, "Diameter Quality-of-Service Application",

RFC 5866, May 2010.

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

[RFC5889] Baccelli, E. and M. Townsley, "IP Addressing Model in Ad Hoc Networks", RFC 5889, September 2010.

[RFC5982] Kobayashi, A. and B. Claise, "IP Flow Information Export (IPFIX) Mediation: Problem Statement", RFC 5982, August 2010.

[RFC5996] Kaufman, C., Hoffman, P., Nir, Y., and P. Eronen, "Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 5996, September 2010.

[RFC6012] Salowey, J., Petch, T., Gerhards, R., and H. Feng, "Datagram Transport Layer Security (DTLS) Transport Mapping for Syslog", RFC 6012, October 2010.

[RFC6020] Bjorklund, M., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, October 2010.

[RFC6021] Schoenwaelder, J., "Common YANG Data Types", RFC 6021, October 2010.

[RFC6022] Scott, M. and M. Bjorklund, "YANG Module

for NETCONF Monitoring", RFC 6022, October 2010.

[RFC6035] Pendleton, A., Clark, A., Johnston, A., and H. Sinnreich, "Session Initiation Protocol Event Package for Voice Quality Reporting", RFC 6035, November 2010.

[RFC6065] Narayan, K., Nelson, D., and R. Presuhn, "Using Authentication, Authorization, and Accounting Services to Dynamically Provision View-Based Access Control Model User-to-Group Mappings", RFC 6065, December 2010.

[RFC6087] Bierman, A., "Guidelines for Authors and Reviewers of YANG Data Model Documents", RFC 6087, January 2011.

[RFC6095] Linowski, B., Ersue, M., and S. Kuryla, "Extending YANG with Language Abstractions", RFC 6095, March 2011.

[RFC6110] Lhotka, L., "Mapping YANG to Document Schema Definition Languages and Validating NETCONF Content", RFC 6110, February 2011.

[RFC6158] DeKok, A. and G. Weber, "RADIUS Design Guidelines", BCP 158, RFC 6158, March 2011.

[RFC6183] Kobayashi, A., Claise, B., Muenz, G., and K.

Ishibashi, "IP Flow Information Export (IPFIX) Mediation: Framework", RFC 6183, April 2011.

[RFC6235] Boschi, E. and B. Trammell, "IP Flow Anonymization Support", RFC 6235, May 2011.

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

[RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", RFC 6242, June 2011.

[RFC6244] Shafer, P., "An Architecture for Network Management Using NETCONF and YANG", RFC 6244, June 2011.

[RFC6248] Morton, A., "RFC 4148 and the IP Performance Metrics (IPPM) Registry of Metrics Are Obsolete", RFC 6248, April 2011.

[RFC6272] Baker, F. and D. Meyer, "Internet Protocols for the Smart Grid", RFC 6272, June 2011.

[RFC6313] Claise, B., Dhandapani, G., Aitken, P., and S. Yates, "Export of Structured Data in IP Flow Information Export (IPFIX)", RFC 6313, July 2011.

[RFC6320] Wadhwa, S., Moisand, J., Haag, T., Voigt, N., and T. Taylor, "Protocol for Access Node Control Mechanism in Broadband Networks", RFC 6320, October 2011.

[RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, January 2012.

[RFC6353] Hardaker, W., "Transport Layer Security (TLS) Transport Model for the Simple Network Management Protocol (SNMP)", RFC 6353, July 2011.

[RFC6371] Busi, I. and D. Allan, "Operations, Administration, and Maintenance Framework for MPLS-Based Transport Networks", RFC 6371, September 2011.

[RFC6408] Jones, M., Korhonen, J., and L. Morand, "Diameter Straightforward-Naming Authority Pointer (S-NAPTR) Usage", RFC 6408, November 2011.

[RFC6536] Bierman, A. and M. Bjorklund, "Network Configuration Protocol (NETCONF) Access Control Model", RFC 6536, March 2012.

[RFCSEARCH] IETF, "RFC Index Search Engine", January 2006, <http://www.rfc-editor.org/rfcsearch.html>.

[SMI-NUMBERS] IANA, "Network Management Parameters - IANA SMI OID List", March 2012, <http://www.iana.org/assignments/smi-numbers>.

[STD16] Rose, M. and K. McCloghrie, "Structure and Identification of Management Information for TCP/IP-based Internets", May 1990.

[STD17] McCloghrie, K. and M. Rose, "Management Information Base for Network Management of TCP/IP-based internets: MIB-II", March 1991.

[STD58] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Structure of Management Information Version 2 (SMIv2)", April 1999.

[STD59] Waldbusser, S., "Remote Network Monitoring Management Information Base", May 2000.

[STD6] Postel, J., "User Datagram Protocol", August 1980.

[STD62] Harrington, D., "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks", December 2002.

[STD66] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax",

January 2005.

[STD7] Postel, J., "Transmission Control Protocol", September 1981.

[XPATH] World Wide Web Consortium, "XML Path Language (XPath) Version 1.0", November 1999, <http://www.w3.org/TR/1999/REC-xpath-19991116>.

[XSD-1] Beech, D., Thompson, H., Maloney, M., Mendelsohn, N., and World Wide Web Consortium Recommendation REC-xmlschema-1-20041028, "XML Schema Part 1: Structures Second Edition", October 2004, <http://www.w3.org/TR/2004/REC-xmlschema-1-20041028>.

Appendix A. High Level Classification of Management Protocols and Data Models

The following subsections aim to guide the reader for the fast selection of the management standard in interest and can be used as a dispatcher to forward to the appropriate chapter. The subsections below classify the protocols on one hand according to high-level criteria such as push versus pull mechanism, and passive versus active monitoring. On the other hand, the protocols are categorized concerning the network management task they address or the data model extensibility they provide. Based on the reader's requirements a reduced set of standard protocols and associated data models can be selected for further reading.

As an example, someone outside of IETF typically would look for the TWAMP protocol in the Operations and Management Area working groups as it addresses performance management. However, the protocol TWAMP has been developed by the IPPM working group in the Transport Area.

Note that not all protocols have been listed in all classification sections. Some of the protocols, especially the protocols with specific focus in Section 3 cannot be clearly classified. Note also

that COPS and COPS-PR are not listed in the tables, as COPS-PR is not recommended to use (see Section 3.3).

A.1. Protocols classified by the Standard Maturity at IETF

This section classifies the management protocols according their standard maturity at the IETF. The IETF standard maturity levels Proposed, Draft or Internet Standard, are defined in [RFC2026]. An Internet Standard is characterized by a high degree of technical maturity and by a generally held belief that the specified protocol or service provides significant benefit to the Internet community.

The table below covers the standard maturity of the different protocols listed in this document. Note that only the main protocols (and not their extensions) are noted. An RFC search tool listing the current document status is available at [RFCSEARCH].

Protocol	Maturity Level
SNMP [STD62][RFC3411] (Section 2.1)	Internet Standard
SYSLOG [RFC5424] (Section 2.2)	Proposed Standard
IPFIX [RFC5101] (Section 2.3)	Proposed Standard
PSAMP [RFC5476] (Section 2.3)	Proposed Standard
NETCONF [RFC6241] (Section 2.4.1)	Proposed Standard
DHCP for IPv4 [RFC2131] (Section 3.1.1)	Draft Standard
DHCP for IPv6 [RFC3315] (Section 3.1.1)	Proposed Standard
OWAMP [RFC4656] (Section 3.4)	Proposed Standard
TWAMP [RFC5357] (Section 3.4)	Proposed Standard
RADIUS [RFC2865] (Section 3.5)	Draft Standard
DIAMETER [RFC3588] (Section 3.6)	Proposed Standard
CAPWAP [RFC5416] (Section 3.7)	Proposed Standard
ANCP [RFC6320] (Section 3.8)	Proposed Standard
Ad-hoc network configuration [RFC5889] (Section 3.1.2)	Informational
ACAP [RFC2244] (Section 3.9)	Proposed Standard

XCAP [RFC4825] (Section 3.10)	Proposed Standard

Table 1: Protocols classified by Standard Maturity at IETF

A.2. Protocols Matched to Management Tasks

This subsection classifies the management protocols matching to the management tasks for fault, configuration, accounting, performance, and security management.

Fault Mgmt	Configuration Mgmt	Accounting Mgmt	Performance Mgmt	Security Mgmt
SNMP notification with trap operation (S. 2.1.1)	SNMP configuration with set operation (S. 2.1.1)	SNMP monitoring with get operation (S. 2.1.1)	SNMP monitoring with get operation (S. 2.1.1)	
IPFIX (S. 2.3)	CAPWAP (S. 3.7)	IPFIX (S. 2.3)	IPFIX (S. 2.3)	
PSAMP (S. 2.3)	NETCONF (S. 2.4)	PSAMP (S. 2.3)	PSAMP (S. 2.3)	
SYSLOG (S. 2.2)	ANCP (S. 3.8)	RADIUS Accounting (S. 3.5)		RADIUS Authent.& Authoriz. (S. 3.5)
	AUTOCONF (S. 3.1.2)	DIAMETER Accounting (S. 3.6)		DIAMETER Authent.& Authoriz. (S. 3.6)
	ACAP (S. 3.9)			
	XCAP (S. 3.10)			
	DHCP (S. 3.11)			

Table 2: Protocols Matched to Management Tasks

Note: Corresponding section numbers are given in parenthesis.

A.3. Push versus Pull Mechanism

A pull mechanism is characterized by the Network Management System (NMS) pulling the management information out of network elements, when needed. A push mechanism is characterized by the network elements pushing the management information to the NMS, either when the information is available, or on a regular basis.

Client/Server protocols, such as DHCP, ANCP, ACAP, and XCAP are not listed in Table 3.

Protocols supporting the Pull mechanism	Protocols supporting the Push mechanism
SNMP (except notifications) (Section 2.1)	SNMP notifications (Section 2.1)
NETCONF (except notifications) (Section 2.4.1)	NETCONF notifications (Section 2.4.1)
CAPWAP (Section 3.7)	SYSLOG (Section 2.2)
	IPFIX (Section 2.3)
	PSAMP (Section 2.3)
	RADIUS accounting (Section 3.5)
	DIAMETER accounting (Section 3.6)

Table 3: Protocol classification by Push versus Pull Mechanism

A.4. Passive versus Active Monitoring

Monitoring can be divided into two categories, passive and active monitoring. Passive monitoring can perform the network traffic monitoring, monitoring of a device or the accounting of network resource consumption by users. Active monitoring, as used in this document, focuses mainly on active network monitoring and relies on the injection of specific traffic (also called "synthetic traffic"), which is then monitored. The monitoring focus is indicated in the table below as "network", "device" or "accounting".

This classification excludes non-monitoring protocols, such as configuration protocols: Ad-hoc network autoconfiguration, ANCP, and XCAP. Note that some of the active monitoring protocols, in the context of the data path, e.g. ICMP Ping and Traceroute [RFC1470], Bidirectional Forwarding Detection (BFD) [RFC5880], and PWE3 Virtual Circuit Connectivity Verification (VCCV) [RFC5085] are covered in [I-D.ietf-opsawg-oam-overview].

Protocols supporting passive monitoring	Protocols supporting active monitoring
IPFIX (network) (Section 2.3)	OWAMP (network) (Section 3.4)
PSAMP (network) (Section 2.3)	TWAMP (network) (Section 3.4)
SNMP (network and device) (Section 2.1)	
NETCONF (device) (Section 2.4.1)	
RADIUS (accounting) (Section 3.5)	
DIAMETER (accounting) (Section 3.6)	
CAPWAP (device) (Section 3.7)	

Table 4: Protocols for passive and active monitoring and their monitoring focus

The application of SNMP to passive traffic monitoring (e.g. with RMON-MIB) or active monitoring (with IPPM MIB) depends on the MIB modules used. However, SNMP protocol itself does not have operations, which support active monitoring. NETCONF can be used for passive monitoring, e.g. with the NETCONF Monitoring YANG module [RFC6022] for the monitoring of the NETCONF protocol. CAPWAP monitors the status of a Wireless Termination Point.

RADIUS and DIAMETER are considered as passive monitoring protocols as they perform accounting, i.e. counting the number of packets/bytes for a specific user.

A.5. Supported Data Model Types and their Extensibility

The following table matches the protocols to the associated data model types. Furthermore, the table indicates how the data model can be extended based on the available content today and whether the protocol contains a built-in mechanism for proprietary extensions of the data model.

Protocol	Data Modeling	Data Model Extensions	Proprietary Data Modeling Extensions
SNMP (Section 2.1)	MIB modules defined with SMI (Section 2.1.3)	New MIB modules specified in new RFCs	Enterprise specific MIB modules
SYSLOG (Section 2.2)	Structured Data Elements (SDE) (Section 4.2.1)	With the procedure to add Structured Data ID in [RFC5424]	Enterprise specific SDEs
IPFIX (Section 2.3)	IPFIX Information Elements, IPFIX IANA registry at [IANA-IPFIX] (Section 2.3)	With the procedure to add Information Elements specified in [RFC5102]	Enterprise specific Information Elements [RFC5101]
PSAMP (Section 2.3)	PSAMP Information Elements, as an extension to IPFIX [IANA-IPFIX], and PSAMP IANA registry at [IANA-PSAMP] (Section 2.3)	With the procedure to add Information Elements specified in [RFC5102]	Enterprise specific Information Elements [RFC5101]
NETCONF (Section 2.4.1)	YANG modules (Section 2.4.2)	New YANG modules specified in new RFCs following the guideline in [RFC6087]	Enterprise specific YANG modules
IPPM OWAMP/TWAMP (Section 3.4)	IPPM metrics (*) (Section 3.4)	New IPPM metrics (Section 3.4)	Not applicable
RADIUS (Section 3.5)	Type-Length-Values (TLV)	RADIUS related registries at [IANA-AAA] and [IANA-PROT]	Vendor Specific Attributes [RFC2865]

DIAMETER (Section 3.6)	Attribute-Value Pairs (AVP)	DIAMETER related registry at [IANA-AAA]	Vendor Specific Attributes [RFC2865]
CAPWAP (Section 3.7)	Type-Length-Values (TLV)	New bindings specified in new RFCs	Vendor specific TLVs

Table 5: Data Models and their Extensibility

(*): With the publication of [RFC6248] the latest IANA registry for IPFIX metrics has been declared Obsolete.

Appendix B. New Work related to IETF Management Standards

B.1. Energy Management (EMAN)

Energy management is becoming an additional requirement for network management systems due to several factors including the rising and fluctuating energy costs, the increased awareness of the ecological impact of operating networks and devices, and the regulation of governments on energy consumption and production.

The basic objective of energy management is operating communication networks and other equipments with a minimal amount of energy while still providing sufficient performance to meet service level objectives. Today, most networking and network-attached devices neither monitor nor allow control energy usage as they are mainly instrumented for functions such as fault, configuration, accounting, performance, and security management. These devices are not instrumented to be aware of energy consumption. There are very few means specified in IETF documents for energy management, which includes the areas of power monitoring, energy monitoring, and power state control.

A particular difference between energy management and other management tasks is that in some cases energy consumption of a device is not measured at the device itself but reported by a different place. For example, at a Power over Ethernet (PoE) sourcing device or at a smart power strip, where one device is effectively metering another remote device. This requires a clear definition of the relationship between the reporting devices and identification of remote devices for which monitoring information is provided. Similar considerations will apply to power state control of remote devices, for example, at a PoE sourcing device that switches on and off power at its ports. Another example scenario for energy management is a gateway to low resourced and lossy network devices in wireless a

building network. Here the energy management system talks directly to the gateway but not necessarily to other devices in the building network.

At the time of this writing the EMAN working group works on the management of energy-aware devices, covered by the following items:

- Requirements for energy management, specifying energy management properties that will allow networks and devices to become energy aware. In addition to energy awareness requirements, the need for control functions will be discussed. Specifically the need to monitor and control properties of devices that are remote to the reporting device should be discussed.

- Energy management framework, which will describe extensions to current management framework, required for energy management. This includes: power and energy monitoring, power states, power state control, and potential power state transitions. The framework will focus on energy management for IP-based network equipment (routers, switches, PCs, IP cameras, phones and the like). Particularly, the relationships between reporting devices, remote devices, and monitoring probes (such as might be used in low-power and lossy networks) need to be elaborated. For the case of a device reporting on behalf of other devices and controlling those devices, the framework will address the issues of discovery and identification of remote devices.

- Energy-aware Networks and Devices MIB document, for monitoring energy-aware networks and devices, will address devices identification, context information, and potential relationship between reporting devices, remote devices, and monitoring probes.

- Power and Energy Monitoring MIB document will document defining managed objects for monitoring of power states and energy consumption/production. The monitoring of power states includes: retrieving power states, properties of power states, current power state, power state transitions, and power state statistics. The managed objects will provide means of reporting detailed properties of the actual energy rate (power) and of accumulated energy. Further, it will provide information on electrical power quality.

- Battery MIB document will define managed objects for battery monitoring, which will provide means of reporting detailed properties of the actual charge, age, and state of a battery and of battery statistics.

- Applicability statement will describe the variety of applications that can use the energy framework and associated MIB modules. Potential examples are building networks, home energy gateway, etc. Finally, the document will also discuss relationships of the framework to other architectures and frameworks (such as Smart Grid). The applicability statement will explain the relationship between the work in this WG and other existing standards e.g. from the IEC, ANSI, DMTF, etc. Note that the EMAN WG will be looking into existing standards such as those from the IEC, ANSI, DMTF and others, and reuse existing work as much as possible.

Appendix C. Change Log

RFC EDITOR: Please remove this appendix before publication.

C.1. 06-07

- Addressed IESG requests.

C.2. 05-06

- Added a description for each DIAMETER application.
- Extend text for XCAP and added descriptions for XCAP application usages.
- Addressed GEN-area review comments.
- Fixed nits and references.

C.3. 04-05

- Fixed nits.

C.4. 03-04

- Resolved many bugs, nits and open issues.
- Reduced text on old and less used RFCs.
- Formulated text on drafts, which are not expected to be published in IETF 83 time frame, as ongoing work and deleted the reference.
- Reduced I-D references and edited remaining ones as easily replaceable with RFC references.
- Removed textual references that RFCs are Proposed or Draft standard.

- Removed the categories for Draft, Proposed and Full standards in section 4.2.

C.5. 02-03

- Added the new subsection 4.1 giving a broader overview of IETF management data models.
- Reduced text on RMON in section 4.2.4 Performance Management
- Resolved bugs, nits and open issues
- Added RFC references

C.6. 01-02

- Resolved bugs, nits and open issues
- Reduced subsections RADIUS and DIAMETER with text on expired drafts.
- Extended dispatcher tables in Appendix A
- Added a note indicating that IETF has not developed so far specific technologies for the management of sensor networks.
- Added a note that IETF has not used the FCAPS view as an organizing principle for its data models.
- Added draft-weil-shared-transition-space-request assuming that it’ll get published pretty fast
- Added RFC references
- Removed text on expired drafts

C.7. 00-01

- Reduced text for the Security Requirements on SNMP and referenced to RFC 3411
- Reduced subsection on VACM
- Merged subsection on "RADIUS Authentication and Authorization with SNMP Transport Models" into the section "SNMP Transport Security Model"

- Section on Dynamic Host Configuration Protocol (DHCP) revised by Ralph Droms
- Subsections on DHCP and Autoconf assembled in section "IP Address Management"
- Removed subsection on "Extensible Provision Protocol (EPP)"
- Introduced new Appendix on "High Level Classification of Management Protocols and Data Models"
- Deleted detailed positive comments
- Resolved some of the I-D references with the correct reference to the published RFC number
- Added RFC references
- Removed text on expired drafts
- Resolved bugs, nits and open issues

C.8. draft-ersue-opsawg-management-fw-03-00

- Diverse bug fixing
- Incorporated comments from Juergen Schoenwaelder
- Reduced detailed text on pro and contra on management technologies
- Extended Terminology section with terms and abbreviations
- Explained the structure based on the management application view
- Definition of ’MIB module’ aligned in different sections
- Text on SNMP security reduced
- All protocol sections discuss now security and AAA as far as relevant
- Added IPFIX IEs, SYSLOG SDEs, and YANG modules to the data model definition
- Added text on YANG data modules to section 4.2.
- Added text on IPFIX IEs to section 4.3.

- Added numerous references

C.9. Change Log from draft-ersue-opsawg-management-fw

C.9.1. 02-03

- Rearranged the document structure using a flat structure putting all protocols onto the same level.
- Incorporated contributions for RADIUS/DIAMETER, IPFIX/PSAMP, YANG, and EMAN.
- Added diverse references.
- Added Contributors and Acknowledgements sections.
- Bug fixing and issue solving.

C.9.2. 01-02

- Added terminology section.
- Changed the language for neutral standard description addressing diverse SDOs.
- Extended NETCONF and NETMOD related text.
- Extended section for 'IPv6 Network Operations'.
- Bug fixing.

C.9.3. 00-01

- Extended text for SNMP
- Extended RADIUS and DIAMETER sections.
- Added references.
- Bug fixing.

Authors' Addresses

Mehmet Ersue (editor)
Nokia Siemens Networks
St.-Martin-Strasse 53
Munich 81541
Germany

EMail: mehmet.ersue@nsn.com

Benoit Claise
Cisco Systems, Inc.
De Kleetlaan 6a b1
Diegem 1831
Belgium

EMail: bclaise@cisco.com

Operations and Management Area Working Group
Internet Draft
Intended status: Informational
Expires: September 2014

T. Mizrahi
Marvell
N. Sprecher
Nokia Solutions and Networks
E. Bellagamba
Ericsson
Y. Weingarten

March 28, 2014

An Overview of Operations, Administration, and Maintenance (OAM) Tools

draft-ietf-opsawg-oam-overview-16.txt

Abstract

Operations, Administration, and Maintenance (OAM) is a general term that refers to a toolset for fault detection and isolation, and for performance measurement. Over the years various OAM tools have been defined for various layers in the protocol stack.

This document summarizes some of the OAM tools defined in the IETF in the context of IP unicast, MPLS, MPLS Transport Profile (MPLS-TP), pseudowires, and TRILL. This document focuses on tools for detecting and isolating failures in networks and for performance monitoring. Control and management aspects of OAM are outside the scope of this document. Network repair functions such as Fast Reroute (FRR) and protection switching, which are often triggered by OAM protocols, are also out of the scope of this document.

The target audience of this document includes network equipment vendors, network operators and standards development organizations, and can be used as an index to some of the main OAM tools defined in the IETF. This document provides a brief description of each of the OAM tools in the IETF. At the end of the document a list of the OAM toolsets and a list of the OAM functions are presented as a summary.

This Internet-Draft will expire on September 28, 2014.

Table of Contents

1. Introduction

OAM is a general term that refers to a toolset for detecting, isolating and reporting failures and for monitoring the network performance.

There are several different interpretations to the "OAM" acronym. This document refers to Operations, Administration and Maintenance, as recommended in Section 3 of [OAM-Def].

This document summarizes some of the OAM tools defined in the IETF in the context of IP unicast, MPLS, MPLS Transport Profile (MPLS-TP), pseudowires, and TRILL.

This document focuses on tools for detecting and isolating failures and for performance monitoring. Hence, this document focuses on the tools used for monitoring and measuring the data plane; control and management aspects of OAM are outside the scope of this document. Network repair functions such as Fast Reroute (FRR) and protection switching, which are often triggered by OAM protocols, are also out of the scope of this document.

1.1. Background

OAM was originally used in traditional communication technologies such as E1 and T1, evolving into PDH and then later in SONET/SDH. ATM was probably the first technology to include inherent OAM support from day one, while in other technologies OAM was typically defined in an ad hoc manner after the technology was already defined and deployed. Packet-based networks were traditionally considered unreliable and best-effort. As packet-based networks evolved, they have become the common transport for both data and telephony, replacing traditional transport protocols. Consequently, packet-based networks were expected to provide a similar "carrier grade" experience, and specifically to support more advanced OAM functions, beyond ICMP and router hellos, that were traditionally used for fault detection.

As typical networks have a multi-layer architecture, the set of OAM protocols similarly take a multi-layer structure; each layer has its

own OAM protocols. Moreover, OAM can be used at different levels of hierarchy in the network to form a multi-layer OAM solution, as shown in the example in Figure 1.

Figure 1 illustrates a network in which IP traffic between two customer edges is transported over an MPLS provider network. MPLS OAM is used at the provider-level for monitoring the connection between the two provider edges, while IP OAM is used at the customer-level for monitoring the end-to-end connection between the two customer edges.

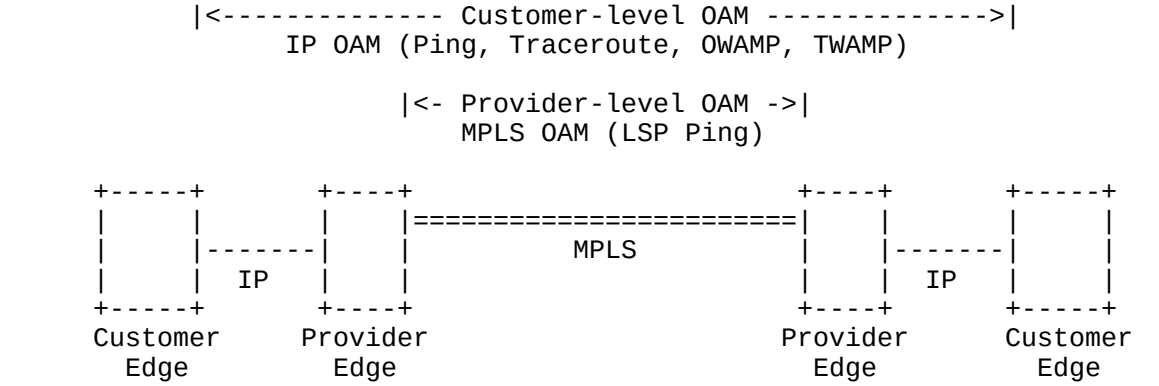

Figure 1 Example: Multi-layer OAM

1.2. Target Audience

The target audience of this document includes:

- Standards development organizations - both IETF working groups and non-IETF organizations can benefit from this document when designing new OAM protocols, or when looking to reuse existing OAM tools for new technologies.

- Network equipment vendors and network operators - can use this document as an index to some of the common IETF OAM tools.

It should be noted that some background in OAM is necessary in order to understand and benefit from this document. Specifically, the reader is assumed to be familiar with the term OAM [OAM-Def], the motivation for using OAM, and the distinction between OAM and network management [OAM-Mng].

1.3. OAM-related Work in the IETF

This memo provides an overview of the different sets of OAM tools defined by the IETF. The set of OAM tools described in this memo are applicable to IP unicast, MPLS, pseudowires, MPLS Transport Profile (MPLS-TP), and TRILL. While OAM tools that are applicable to other technologies exist, they are beyond the scope of this memo.

This document focuses on IETF documents that have been published as RFCs, while other ongoing OAM-related work is outside the scope.

The IETF has defined OAM protocols and tools in several different contexts. We roughly categorize these efforts into a few sets of OAM-related RFCs, listed in Table 1. Each set defines a logically-coupled set of RFCs, although the sets are in some cases intertwined by common tools and protocols.

The discussion in this document is ordered according to these sets (the acronyms and abbreviations are listed in Section 2.1.).

Toolset	Transport Technology
IP Ping	IPv4/IPv6
IP Traceroute	IPv4/IPv6
BFD	generic
MPLS OAM	MPLS
MPLS-TP OAM	MPLS-TP
Pseudowire OAM	Pseudowires
OWAMP and TWAMP	IPv4/IPv6
TRILL OAM	TRILL

Table 1 OAM Toolset Packages in the IETF Documents

This document focuses on OAM tools that have been developed in the IETF. A short summary of some of the significant OAM standards that have been developed in other standard organizations is presented in Appendix A.2.

1.4. Focusing on the Data Plane

OAM tools may, and quite often do, work in conjunction with a control plane and/or management plane. OAM provides instrumentation tools for measuring and monitoring the data plane. OAM tools often use control plane functions, e.g., to initialize OAM sessions and to exchange various parameters. The OAM tools communicate with the management plane to raise alarms, and often OAM tools may be activated by the management (as well as by the control plane), e.g., to locate and localize problems.

The considerations of the control plane maintenance tools and the functionality of the management plane are out of scope for this document, which concentrates on presenting the data plane tools that are used for OAM. Network repair functions such as Fast Reroute (FRR) and protection switching, which are often triggered by OAM protocols, are also out of the scope of this document.

Since OAM protocols are used for monitoring the data plane, it is imperative for OAM tools to be capable of testing the actual data plane with as much accuracy as possible. Thus, it is important to enforce fate-sharing between OAM traffic that monitors the data plane and the data plane traffic it monitors.

2. Terminology

2.1. Abbreviations

ACH Associated Channel Header

AIS Alarm Indication Signal

ATM Asynchronous Transfer Mode

BFD Bidirectional Forwarding Detection

CC Continuity Check

CV Connectivity Verification

DM Delay Measurement

ECMP Equal Cost Multiple Paths

FEC Forwarding Equivalence Class

FRR Fast Reroute

G-ACh Generic Associated Channel

GAL Generic Associated Label

ICMP Internet Control Message Protocol

L2TP Layer Two Tunneling Protocol

L2VPN Layer Two Virtual Private Network

L3VPN Layer Three Virtual Private Network

LCCE L2TP Control Connection Endpoint

LDP Label Distribution Protocol

LER Label Edge Router

LM Loss Measurement

LSP Label Switched Path

LSR Label Switched Router

ME Maintenance Entity

MEG Maintenance Entity Group

MEP MEG End Point

MIP MEG Intermediate Point

MP Maintenance Point

MPLS Multiprotocol Label Switching

MPLS-TP MPLS Transport Profile

MTU Maximum Transmission Unit

OAM Operations, Administration, and Maintenance

OWAMP One-way Active Measurement Protocol

PDH Plesiochronous Digital Hierarchy

PE Provider Edge

PSN Public Switched Network

PW Pseudowire

PWE3 Pseudowire Emulation Edge-to-Edge

RBridge Routing Bridge

RDI Remote Defect Indication

SDH Synchronous Digital Hierarchy

SONET Synchronous Optical Networking

TRILL Transparent Interconnection of Lots of Links

TTL Time To Live

TWAMP Two-way Active Measurement Protocol

VCCV Virtual Circuit Connectivity Verification

VPN Virtual Private Network

2.2. Terminology used in OAM Standards

2.2.1. General Terms

A wide variety of terms is used in various OAM standards. This section presents a comparison of the terms used in various OAM standards, without fully quoting the definition of each term.

An interesting overview of the term OAM and its derivatives is presented in [OAM-Def]. A thesaurus of terminology for MPLS-TP terms is presented in [TP-Term], and provides a good summary of some of the OAM related terminology.

2.2.2. Operations, Administration and Maintenance

The following definition of OAM is quoted from [OAM-Def]:

The components of the "OAM" acronym (and provisioning) are defined as follows:

- Operations - Operation activities are undertaken to keep the network (and the services that the network provides) up and running. It includes monitoring the network and finding problems. Ideally these problems should be found before users are affected.

- Administration - Administration activities involve keeping track of resources in the network and how they are used. It includes all the bookkeeping that is necessary to track networking resources and the network under control.

- Maintenance - Maintenance activities are focused on facilitating repairs and upgrades -- for example, when equipment must be replaced, when a router needs a patch for an operating system image, or when a new switch is added to a network. Maintenance also involves corrective and preventive measures to make the managed network run more effectively, e.g., adjusting device configuration and parameters.

2.2.3. Functions, Tools and Protocols

OAM Function

An OAM function is an instrumentation measurement type or diagnostic.

OAM functions are the atomic building blocks of OAM, where each function defines an OAM capability.

Typical examples of OAM functions are presented in Section 3.

OAM Protocol

A protocol used for implementing one or more OAM functions.

The OWAMP-Test [OWAMP] is an example of an OAM protocol.

OAM Tool

An OAM tool is a specific means of applying one or more OAM functions.

In some cases an OAM protocol *is* an OAM tool, e.g., OWAMP-Test. In other cases an OAM tool uses a set of protocols that are not strictly OAM-related; for example, Traceroute (Section 4.2.) can be

implemented using UDP and ICMP messages, without using an OAM protocol per se.

2.2.4. Data Plane, Control Plane and Management Plane

Data Plane

The data plane is the set of functions used to transfer data in the stratum or layer under consideration [ITU-Terms].

The Data Plane is also known as the Forwarding Plane or the User Plane.

Control Plane

The control plane is the set of protocols and mechanisms that enable routers to efficiently learn how to forward packets towards their final destination (based on [Comp]).

Management Plane

The term Management Plane, as described in [Mng], is used to describe the exchange of management messages through management protocols (often transported by IP and by IP transport protocols) between management applications and the managed entities such as network nodes.

Data Plane vs. Control Plane vs. Management Plane

The distinction between the planes is at times a bit vague. For example, the definition of "Control Plane" above may imply that OAM tools such as ping, BFD and others are in fact in the control plane.

This document focuses on tools used for monitoring the data plane. While these tools could arguably be considered to be in the control plane, these tools monitor the data plane, and hence it is imperative to have fate-sharing between OAM traffic that monitors the data plane and the data plane traffic it monitors.

Another potentially vague distinction is between the management plane and control plane. The management plane should be seen as separate from, but possibly overlapping with, the control plane (based on [Mng]).

2.2.5. The Players

An OAM tool is used between two (or more) peers. Various terms are used in IETF documents to refer to the players that take part in OAM. Table 2 summarizes the terms used in each of the toolsets discussed in this document.

Toolset	Terms
Ping / Traceroute ([ICMPv4], [ICMPv6], [TCPIP-Tools])	-Host -Node -Interface -Gateway
BFD [BFD]	System
MPLS OAM [MPLS-OAM-FW]	LSR
MPLS-TP OAM [TP-OAM-FW]	-End Point - MEP -Intermediate Point - MIP
Pseudowire OAM [VCCV]	-PE -LCCE
OWAMP and TWAMP ([OWAMP], [TWAMP])	-Host -End system
TRILL OAM [TRILL-OAM]	-RBridge

Table 2 Maintenance Point Terminology

2.2.6. Proactive and On-demand Activation

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

Proactive

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

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

2.2.7. Connectivity Verification and Continuity Checks

Two distinct classes of failure management functions are used in OAM protocols, connectivity verification and continuity checks. The distinction between these terms is defined in [MPLS-TP-OAM], and is used similarly in this document.

Continuity Check

Continuity checks are used to verify that a destination is reachable, and are typically sent proactively, though they can be invoked on-demand as well.

Connectivity Verification

A connectivity verification function allows Alice to check whether she is connected to Bob or not. It is noted that while the CV function is performed in the data plane, the "expected path" is predetermined either in the control plane or in the management plane. A connectivity verification (CV) protocol typically uses a CV message, followed by a CV reply that is sent back to the originator. A CV function can be applied proactively or on-demand.

Connectivity verification tools often perform path verification as well, allowing Alice to verify that messages from Bob are received through the correct path, thereby verifying not only that the two MPs are connected, but also that they are connected through the expected path, allowing detection of unexpected topology changes.

Connectivity verification functions can also be used for checking the MTU of the path between the two peers.

Connectivity verification and continuity checks are considered complementary mechanisms, and are often used in conjunction with each other.

2.2.8. Connection Oriented vs. Connectionless Communication

Connection Oriented

In Connection Oriented technologies an end-to-end connection is established (by a control protocol or provisioned by a management system) prior to the transmission of data.

Typically a connection identifier is used to identify the connection. In connection oriented technologies it is often the case (although not always) that all packets belonging to a specific connection use the same route through the network.

Connectionless

In Connectionless technologies data is typically sent between end points without prior arrangement. Packets are routed independently based on their destination address, and hence different packets may be routed in a different way across the network.

Discussion

The OAM tools described in this document include tools that support connection oriented technologies, as well as tools for connectionless technologies.

In connection oriented technologies OAM is used to monitor a *specific* connection; OAM packets are forwarded through the same route as the data traffic and receive the same treatment. In connectionless technologies, OAM is used between a source and destination pair without defining a specific connection. Moreover, in some cases the route of OAM packets may differ from the one of the data traffic. For example, the connectionless IP Ping (Section 4.1.) tests the reachability from a source to a given destination, while the connection oriented LSP Ping (Section 4.4.) is used for monitoring a specific LSP (connection), and provides the capability to monitor all the available paths used by an LSP.

It should be noted that in some cases connectionless protocols are monitored by connection oriented OAM protocols. For example, while IP is a connectionless protocol, it can monitored by BFD (Section 4.3.), which is connection oriented.

2.2.9. Point-to-point vs. Point-to-multipoint Services

Point-to-point (P2P)

A P2P service delivers data from a single source to a single destination.

Point-to-multipoint (P2MP)

A P2MP service delivers data from a single source to a one or more destinations (based on [Signal]).

An MP2MP service is a service that delivers data from more than one source to one or more receivers (based on [Signal]).

Note: the two definitions for P2MP and MP2MP are quoted from [Signal]. Although [Signal] describes a specific case of P2MP and MP2MP which is MPLS-specific, these two definitions also apply to non-MPLS cases.

Discussion

The OAM tools described in this document include tools for P2P services, as well as tools for P2MP services.

The distinction between P2P services and P2MP services affects the corresponding OAM tools. A P2P service is typically simpler to monitor, as it consists of a single pair of end points. P2MP and MP2MP services present several challenges. For example, in a P2MP service, the OAM mechanism not only verifies that each of the destinations is reachable from the source, but also verifies that the P2MP distribution tree is intact and loop-free.

2.2.10. Failures

The terms Failure, Fault, and Defect are used interchangeably in the standards, referring to a malfunction that can be detected by a connectivity or a continuity check. In some standards, such as 802.1ag [IEEE802.1Q] , there is no distinction between these terms, while in other standards each of these terms refers to a different type of malfunction.

The terminology used in IETF MPLS-TP OAM is based on the ITU-T terminology, which distinguishes between these three terms in [ITU-T-G.806];

Fault

The term Fault refers to an inability to perform a required action, e.g., an unsuccessful attempt to deliver a packet.

Defect

The term Defect refers to an interruption in the normal operation, such as a consecutive period of time where no packets are delivered successfully.

Failure

The term Failure refers to the termination of the required function. While a Defect typically refers to a limited period of time, a failure refers to a long period of time.

3. OAM Functions

This subsection provides a brief summary of the common OAM functions used in OAM-related standards. These functions are used as building blocks in the OAM standards described in this document.

- Connectivity Verification (CV), Path Verification and Continuity Checks (CC):
  As defined in Section 2.2.7.

- Path Discovery / Fault Localization:
  This function can be used to trace the route to a destination, i.e., to identify the nodes along the route to the destination. When more than one route is available to a specific destination, this function traces one of the available routes. When a failure occurs, this function attempts to detect the location of the failure.
  Note that the term route tracing (or Traceroute) that is used in the context of IP and MPLS, is sometimes referred to as path tracing in the context of other protocols, such as TRILL.

- Performance Monitoring:
  Typically refers to:

  - Loss Measurement (LM) - monitors the packet loss rate.

  - Delay Measurement (DM) - monitors the delay and delay variation (jitter).

4. OAM Tools in the IETF - a Detailed Description

This section presents a detailed description of the sets of OAM-related tools in each of the toolsets in Table 1.

4.1. IP Ping

Ping is a common network diagnosis application for IP networks that uses ICMP. According to [NetTerms], 'Ping' is an abbreviation for Packet internet groper, although the term has been so commonly used that it stands on its own. As defined in [NetTerms], it is a program used to test reachability of destinations by sending them an ICMP echo request and waiting for a reply.

The ICMP Echo request/reply exchange in Ping is used as a continuity check function for the Internet Protocol. The originator transmits an ICMP Echo request packet, and the receiver replies with an Echo reply. ICMP ping is defined in two variants, [ICMPv4] is used for IPv4, and [ICMPv6] is used for IPv6.

Ping can be invoked either to a unicast destination or to a multicast destination. In the latter case, all members of the multicast group send an Echo reply back to the originator.

Ping implementations typically use ICMP messages. UDP Ping is a variant that uses UDP messages instead of ICMP echo messages.

Ping is a single-ended continuity check, i.e., it allows the *initiator* of the Echo request to test the reachability. If it is desirable for both ends to test the reachability, both ends have to invoke Ping independently.

Note that since ICMP filtering is deployed in some routers and firewalls, the usefulness of Ping is sometimes limited in the wider internet. This limitation is equally relevant to Traceroute.

4.2. IP Traceroute

Traceroute ([TCPIP-Tools], [NetTools]) is an application that allows users to discover a path between an IP source and an IP destination.

The most common way to implement Traceroute [TCPIP-Tools] is described as follows. Traceroute sends a sequence of UDP packets to UDP port 33434 at the destination. By default, Traceroute begins by sending three packets (the number of packets is configurable in most Traceroute implementations), each with an IP Time-To-Live (or Hop Limit in IPv6) value of one to the destination. These packets expire as soon as they reach the first router in the path. Consequently, that router sends three ICMP Time Exceeded Messages back to the Traceroute application. Traceroute now sends another three UDP packets, each with the TTL value of 2. These messages cause the second router to return ICMP messages. This process continues, with

ever increasing values for the TTL field, until the packets actually reach the destination. Because no application listens to port 33434 at the destination, the destination returns ICMP Destination Unreachable Messages indicating an unreachable port. This event indicates to the Traceroute application that it is finished. The Traceroute program displays the round-trip delay associated with each of the attempts.

While Traceroute is a tool that finds *a* path from A to B, it should be noted that traffic from A to B is often forwarded through Equal Cost Multiple Paths (ECMP). Paris Traceroute [PARIS] is an extension to Traceroute that attempts to discovers all the available paths from A to B by scanning different values of header fields (such as UDP ports) in the probe packets.

It is noted that Traceroute is an application, and not a protocol. As such, it has various different implementations. One of the most common ones uses UDP probe packets, as described above. Other implementations exist that use other types of probe messages, such as ICMP or TCP.

Note that IP routing may be asymmetric. While Traceroute discovers a path between a source and destination, it does not reveal the reverse path.

A few ICMP extensions ([ICMP-MP], [ICMP-Int]) have been defined in the context of Traceroute. These documents define several extensions, including extensions to the ICMP Destination Unreachable message, that can be used by Traceroute applications.

Traceroute allows path discovery to *unicast* destination addresses. A similar tool [mtrace] was defined for multicast destination addresses, allowing to trace the route that a multicast IP packet takes from a source to a particular receiver.

4.3. Bidirectional Forwarding Detection (BFD)

4.3.1. Overview

While multiple OAM tools have been defined for various protocols in the protocol stack, Bidirectional Forwarding Detection [BFD], defined by the IETF BFD working group, is a generic OAM tool that can be deployed over various encapsulating protocols, and in various medium types. The IETF has defined variants of the protocol for IP ([BFD-IP], [BFD-Multi]), for MPLS LSPs [BFD-LSP], and for pseudowires [BFD-VCCV]. The usage of BFD in MPLS-TP is defined in [TP-CC-CV].

BFD includes two main OAM functions, using two types of BFD packets: BFD Control packets, and BFD Echo packets.

4.3.2. Terminology

BFD operates between *systems*. The BFD protocol is run between two or more systems after establishing a *session*.

4.3.3. BFD Control

BFD supports a bidirectional continuity check, using BFD control packets, that are exchanged within a BFD session. BFD sessions operate in one of two modes:

- Asynchronous mode (i.e., proactive): in this mode BFD control packets are sent periodically. When the receiver detects that no BFD control packets have been received during a predetermined period of time, a failure is reported.

- Demand mode: in this mode, BFD control packets are sent on-demand. Upon need, a system initiates a series of BFD control packets to check the continuity of the session. BFD control packets are sent independently in each direction.

Each of the end-points (referred to as systems) of the monitored path maintains its own session identification, called a Discriminator, both of which are included in the BFD Control Packets that are exchanged between the end-points. At the time of session establishment, the Discriminators are exchanged between the two-end points. In addition, the transmission (and reception) rate is negotiated between the two end-points, based on information included in the control packets. These transmission rates may be renegotiated during the session.

During normal operation of the session, i.e., when no failures have been detected, the BFD session is in the Up state. If no BFD Control packets are received during a period of time called the Detection Time, the session is declared to be Down. The detection time is a function of the pre-configured or negotiated transmission rate, and a parameter called Detect Mult. Detect Mult determines the number of missing BFD Control packets that cause the session to be declared as Down. This parameter is included in the BFD Control packet.

4.3.4. BFD Echo

A BFD echo packet is sent to a peer system, and is looped back to the originator. The echo function can be used proactively, or on-demand.

The BFD echo function has been defined in BFD for IPv4 and IPv6 ([BFD-IP]), but is not used in BFD for MPLS LSPs, PWs, or in BFD for MPLS-TP.

4.4. MPLS OAM

The IETF MPLS working group has defined OAM for MPLS LSPs. The requirements and framework of this effort are defined in [MPLS-OAM-FW] and [MPLS-OAM], respectively. The corresponding OAM tool defined, in this context, is LSP Ping [LSP-Ping]. OAM for P2MP services is defined in [MPLS-P2MP].

BFD for MPLS [BFD-LSP] is an alternative means for detecting data-plane failures, as described below.

4.4.1. LSP Ping

LSP Ping is modeled after the Ping/Traceroute paradigm and thus it may be used in one of two modes:

- "Ping" mode: In this mode LSP Ping is used for end-to-end connectivity verification between two LERs.
- "Traceroute" mode: This mode is used for hop-by-hop fault isolation.

LSP Ping is based on ICMP Ping operation (of data-plane connectivity verification) with additional functionality to verify data-plane vs. control-plane consistency for a Forwarding Equivalence Class (FEC) and also identify Maximum Transmission Unit (MTU) problems.

The Traceroute functionality may be used to isolate and localize MPLS faults, using the Time-to-live (TTL) indicator to incrementally identify the sub-path of the LSP that is successfully traversed before the faulty link or node.

The challenge in MPLS networks is that the traffic of a given LSP may be load balanced across Equal Cost Multiple paths (ECMP). LSP Ping monitors all the available paths of an LSP by monitoring its different Forwarding Equivalence Classes (FEC). Note that MPLS-TP does not use ECMP, and thus does not require OAM over multiple paths.

Another challenge is that an MPLS LSP does not necessarily have a return path; traffic that is sent back from the egress LSR to the ingress LSR is not necessarily sent over an MPLS LSP, but can be sent through a different route, such as an IP route. Thus, responding to an LSP Ping message is not necessarily as trivial as in IP Ping,

where the responder just swaps the source and destination IP addresses. Note that this challenge is not applicable to MPLS-TP, where a return path is always available.

It should be noted that LSP Ping supports unique identification of the LSP within an addressing domain. The identification is checked using the full FEC identification. LSP Ping is extensible to include additional information needed to support new functionality, by use of Type-Length-Value (TLV) constructs. The usage of TLVs is typically handled by the control plane, as it is not easy to implement in hardware.

LSP Ping supports both asynchronous, as well as, on-demand activation.

4.4.2. BFD for MPLS

BFD [BFD-LSP] can be used to detect MPLS LSP data plane failures.

A BFD session is established for each MPLS LSP that is being monitored. BFD Control packets must be sent along the same path as the monitored LSP. If the LSP is associated with multiple FECs, a BFD session is established for each FEC.

While LSP Ping can be used for detecting MPLS data plane failures and for verifying the MPLS LSP data plane against the control plane, BFD can only be used for the former. BFD can be used in conjunction with LSP Ping, as is the case in MPLS-TP (see Section 4.5.4.).

4.4.3. OAM for Virtual Private Networks (VPN) over MPLS

The IETF has defined two classes of VPNs, Layer 2 VPNs (L2VPN) and Layer 3 VPNs (L3VPN). [L2VPN-OAM] provides the requirements and framework for OAM in the context of Layer 2 Virtual Private Networks (L2VPN), and specifically it also defines the OAM layering of L2VPNs over MPLS. [L3VPN-OAM] provides a framework for the operation and management of Layer 3 Virtual Private Networks (L3VPNs).

4.5. MPLS-TP OAM

4.5.1. Overview

The MPLS working group has defined the OAM toolset that fulfills the requirements for MPLS-TP OAM. The full set of requirements for MPLS-TP OAM are defined in [MPLS-TP-OAM], and include both general requirements for the behavior of the OAM tools and a set of operations that should be supported by the OAM toolset. The set of

mechanisms required are further elaborated in [TP-OAM-FW], which describes the general architecture of the OAM system as well as giving overviews of the functionality of the OAM toolset.

Some of the basic requirements for the OAM toolset for MPLS-TP are:

- MPLS-TP OAM must be able to support both an IP based and non-IP based environment. If the network is IP based, i.e., IP routing and forwarding are available, then the MPLS-TP OAM toolset should rely on the IP routing and forwarding capabilities. On the other hand, in environments where IP functionality is not available, the OAM tools must still be able to operate without dependence on IP forwarding and routing.

- OAM packets and the user traffic are required to be congruent (i.e., OAM packets are transmitted in-band) and there is a need to differentiate OAM packets from ordinary user packets in the data plane. Inherent in this requirement is the principle that MPLS-TP OAM be independent of any existing control-plane, although it should not preclude use of the control-plane functionality. OAM packets are identified by the Generic Associated Label (GAL), which is a reserved MPLS label value (13).

4.5.2. Terminology

Maintenance Entity (ME)

The MPLS-TP OAM tools are designed to monitor and manage a Maintenance Entity (ME). An ME, as defined in [TP-OAM-FW], defines a relationship between two points of a transport path to which maintenance and monitoring operations apply.

The term Maintenance Entity (ME) is used in ITU-T Recommendations (e.g., [ITU-T-Y1731]), as well as in the MPLS-TP terminology ([TP-OAM-FW]).

Maintenance Entity Group (MEG)

The collection of one or more MEs that belongs to the same transport path and that are maintained and monitored as a group are known as a Maintenance Entity Group (based on [TP-OAM-FW]).

Maintenance Point (MP)

A Maintenance Point (MP) is a functional entity that is defined at a node in the network, and can initiate and/or react to OAM messages. This document focuses on the data-plane functionality of MPs, while

MPs interact with the control plane and with the management plane as well.

The term MP is used in IEEE 802.1ag, and was similarly adopted in MPLS-TP ([TP-OAM-FW]).

Maintenance End Point (MEP)

A Maintenance End Point (MEP) is one of the end points of an ME, and can initiate OAM messages and respond to them (based on [TP-OAM-FW]).

Maintenance Intermediate Point (MIP)

In between MEPs, there are zero or more intermediate points, called Maintenance Entity Group Intermediate Points (based on [TP-OAM-FW]).

A Maintenance Intermediate Point (MIP) is an intermediate point that does not generally initiate OAM frames (one exception to this is the use of AIS notifications), but is able to respond to OAM frames that are destined to it. A MIP in MPLS-TP identifies OAM packets destined to it by the expiration of the TTL field in the OAM packet. The term Maintenance Point is a general term for MEPs and MIPs.

Up and Down MEPs

The IEEE 802.1ag [IEEE802.1Q] defines a distinction between Up MEPs and Down MEPs. A MEP monitors traffic either in the direction facing the network, or in the direction facing the bridge. A Down MEP is a MEP that receives OAM packets from, and transmits them to the direction of the network. An Up MEP receives OAM packets from, and transmits them to the direction of the bridging entity. MPLS-TP ([TP-OAM-FW]) uses a similar distinction on the placement of the MEP - either at the ingress, egress, or forwarding function of the node (Down / Up MEPs). This placement is important for localization of a failure.

Note that the terms Up and Down MEPs are entirely unrelated to the conventional up/down terminology, where down means faulty, and up is nonfaulty.

The distinction between Up and Down MEPs was defined in [TP-OAM-FW], but has not been used in other MPLS-TP RFCs, as of the writing of this document.

4.5.3. Generic Associated Channel

In order to address the requirement for in-band transmission of MPLS-TP OAM traffic, MPLS-TP uses a Generic Associated Channel (G-ACh), defined in [G-ACh] for LSP-based OAM traffic. This mechanism is based on the same concepts as the PWE3 ACH [PW-ACH] and VCCV [VCCV] mechanisms. However, to address the needs of LSPs as differentiated from PW, the following concepts were defined for [G-ACh]:

- An Associated Channel Header (ACH), that uses a format similar to the PW Control Word [PW-ACH], is a 4-byte header that is prepended to OAM packets.

- A Generic Associated Label (GAL). The GAL is a reserved MPLS label value (13) that indicates that the packet is an ACH packet and the payload follows immediately after the label stack.

It should be noted that while the G-ACh was defined as part of the MPLS-TP definition effort, the G-ACh is a generic tool that can be used in MPLS in general, and not only in MPLS-TP.

4.5.4. MPLS-TP OAM Toolset

To address the functionality that is required of the OAM toolset, the MPLS WG conducted an analysis of the existing IETF and ITU-T OAM tools and their ability to fulfill the required functionality. The conclusions of this analysis are documented in [OAM-Analys]. MPLS-TP uses a mixture of OAM tools that are based on previous standards, and adapted to the requirements of [MPLS-TP-OAM]. Some of the main building blocks of this solution are based on:

- Bidirectional Forwarding Detection ([BFD], [BFD-LSP]) for proactive continuity check and connectivity verification.

- LSP Ping as defined in [LSP-Ping] for on-demand connectivity verification.

- New protocol packets, using G-ACH, to address different functionality.

- Performance measurement protocols that are based on the functionality that is described in [ITU-T-Y1731].

The following sub-sections describe the OAM tools defined for MPLS-TP as described in [TP-OAM-FW].

4.5.4.1. Continuity Check and Connectivity Verification

Continuity Check and Connectivity Verification are presented in Section 2.2.7. of this document. As presented there, these tools may be used either proactively or on-demand. When using these tools proactively, they are generally used in tandem.

For MPLS-TP there are two distinct tools, the proactive tool is defined in [TP-CC-CV] while the on-demand tool is defined in [OnDemand-CV]. In on-demand mode, this function should support monitoring between the MEPs and, in addition, between a MEP and MIP. [TP-OAM-FW] highlights, when performing Connectivity Verification, the need for the CC-V messages to include unique identification of the MEG that is being monitored and the MEP that originated the message.

The proactive tool [TP-CC-CV] is based on extensions to BFD (see Section 4.3.) with the additional limitation that the transmission and receiving rates are based on configuration by the operator. The on-demand tool [OnDemand-CV] is an adaptation of LSP Ping (see Section 4.4.) for the required behavior of MPLS-TP.

4.5.4.2. Route Tracing

[MPLS-TP-OAM] defines that there is a need for functionality that would allow a path end-point to identify the intermediate and end-points of the path. This function would be used in on-demand mode. Normally, this path will be used for bidirectional PW, LSP, and sections, however, unidirectional paths may be supported only if a return path exists. The tool for this is based on the LSP Ping (see Section 4.4.) functionality and is described in [OnDemand-CV].

4.5.4.3. Lock Instruct

The Lock Instruct function [Lock-Loop] is used to notify a transport path end-point of an administrative need to disable the transport path. This functionality will generally be used in conjunction with some intrusive OAM function, e.g., Performance measurement, Diagnostic testing, to minimize the side-effect on user data traffic.

4.5.4.4. Lock Reporting

Lock Reporting is a function used by an end-point of a path to report to its far-end end-point that a lock condition has been affected on the path.

4.5.4.5. Alarm Reporting

Alarm Reporting [TP-Fault] provides the means to suppress alarms following detection of defect conditions at the server sub-layer. Alarm reporting is used by an intermediate point of a path, that becomes aware of a fault on the path, to report to the end-points of the path. [TP-OAM-FW] states that this may occur as a result of a defect condition discovered at a server sub-layer. This generates an Alarm Indication Signal (AIS) that continues until the fault is cleared. The consequent action of this function is detailed in [TP-OAM-FW].

4.5.4.6. Remote Defect Indication

Remote Defect Indication (RDI) is used proactively by a path end-point to report to its peer end-point that a defect is detected on a bidirectional connection between them. [MPLS-TP-OAM] points out that this function may be applied to a unidirectional LSP only if a return path exists. [TP-OAM-FW] points out that this function is associated with the proactive CC-V function.

4.5.4.7. Client Failure Indication

Client Failure Indication (CFI) is defined in [MPLS-TP-OAM] to allow the propagation information from one edge of the network to the other. The information concerns a defect to a client, in the case that the client does not support alarm notification.

4.5.4.8. Performance Monitoring

The definition of MPLS performance monitoring was motivated by the MPLS-TP requirements [MPLS-TP-OAM], but was defined generically for MPLS in [MPLS-LM-DM]. An additional document [TP-LM-DM] defines a performance monitoring profile for MPLS-TP.

4.5.4.8.1. Packet Loss Measurement (LM)

Packet Loss Measurement is a function used to verify the quality of the service. Packet loss, as defined in [IPPM-1LM] and [MPLS-TP-OAM], indicates the ratio of the number of user packets lost to the total number of user packets sent during a defined time interval.

There are two possible ways of determining this measurement:

- Using OAM packets, it is possible to compute the statistics based on a series of OAM packets. This, however, has the disadvantage of being artificial, and may not be representative since part of the packet loss may be dependent upon packet sizes and upon the implementation of the MEPs that take part in the protocol.

- Sending delimiting messages for the start and end of a measurement period during which the source and sink of the path count the packets transmitted and received. After the end delimiter, the ratio would be calculated by the path OAM entity.

4.5.4.8.2. Packet Delay Measurement (DM)

Packet Delay Measurement is a function that is used to measure one-way or two-way delay of a packet transmission between a pair of the end-points of a path (PW, LSP, or Section). Where:

- One-way packet delay, as defined in [IPPM-1DM], is the time elapsed from the start of transmission of the first bit of the packet by a source node until the reception of the last bit of that packet by the destination node. Note that one-way delay measurement requires the clocks of the two end-points to be synchronized.

- Two-way packet delay, as defined in [IPPM-2DM], is the time elapsed from the start of transmission of the first bit of the packet by a source node until the reception of the last bit of the loop-backed packet by the same source node, when the loopback is performed at the packet's destination node. Note that due to possible path asymmetry, the one-way packet delay from one end-point to another is not necessarily equal to half of the two-way packet delay.
  As opposed to one-way delay measurement, two-way delay measurement does not require the two end-points to be synchronized.

For each of these two metrics, the DM function allows the MEP to measure the delay, as well as the delay variation. Delay measurement is performed by exchanging timestamped OAM packets between the participating MEPs.

4.6. Pseudowire OAM

4.6.1. Pseudowire OAM using Virtual Circuit Connectivity Verification (VCCV)

VCCV, as defined in [VCCV], provides a means for end-to-end fault detection and diagnostics tools to be used for PWs (regardless of the

underlying tunneling technology). The VCCV switching function provides a control channel associated with each PW. [VCCV] defines three Control Channel (CC) types, i.e., three possible methods for transmitting and identifying OAM messages:

- o CC Type 1: In-band VCCV, as described in [VCCV], is also referred to as "PWE3 Control Word with 0001b as first nibble". It uses the PW Associated Channel Header [PW-ACH].

- o CC Type 2: Out-of-band VCCV [VCCV], is also referred to as "MPLS Router Alert Label". In this case the control channel is created by using the MPLS router alert label [MPLS-ENCAPS] immediately above the PW label.

- o CC Type 3: TTL expiry VCCV [VCCV], is also referred to as "MPLS PW Label with TTL == 1", i.e., the control channel is identified when the value of the TTL field in the PW label is set to 1.

VCCV currently supports the following OAM tools: ICMP Ping, LSP Ping, and BFD. ICMP and LSP Ping are IP encapsulated before being sent over the PW ACH. BFD for VCCV [BFD-VCCV] supports two modes of encapsulation - either IP/UDP encapsulated (with IP/UDP header) or PW-ACH encapsulated (with no IP/UDP header) and provides support to signal the AC status. The use of the VCCV control channel provides the context, based on the MPLS-PW label, required to bind and bootstrap the BFD session to a particular pseudo wire (FEC), eliminating the need to exchange Discriminator values.

VCCV consists of two components: (1) signaled component to communicate VCCV capabilities as part of VC label, and (2) switching component to cause the PW payload to be treated as a control packet.

VCCV is not directly dependent upon the presence of a control plane. The VCCV capability advertisement may be performed as part of the PW signaling when LDP is used. In case of manual configuration of the PW, it is the responsibility of the operator to set consistent options at both ends. The manual option was created specifically to handle MPLS-TP use cases where no control plane was a requirement. However, new use cases such as pure mobile backhaul find this functionality useful too.

The PWE3 working group has conducted an implementation survey of VCCV [VCCV-SURVEY], which analyzes which VCCV mechanisms are used in practice.

4.6.2. Pseudowire OAM using G-ACh

As mentioned above, VCCV enables OAM for PWs by using a control channel for OAM packets. When PWs are used in MPLS-TP networks, rather than the control channels defined in VCCV, the G-ACh can be used as an alternative control channel. The usage of the G-ACh for PWs is defined in [PW-G-ACh].

4.6.3. Attachment Circuit - Pseudowire Mapping

The PWE3 working group has defined a mapping and notification of defect states between a pseudowire (PW) and the Attachment Circuits (ACs) of the end-to-end emulated service. This mapping is of key importance to the end-to-end functionality. Specifically, the mapping is provided by [PW-MAP], by [L2TP-EC] for L2TPv3 pseudowires, and Section 5.3 of [ATM-L2] for ATM.

[L2VPN-OAM] provides the requirements and framework for OAM in the context of Layer 2 Virtual Private Networks (L2VPN), and specifically it also defines the OAM layering of L2VPNs over pseudowires.

The mapping defined in [Eth-Int] allows an end-to-end emulated Ethernet service over pseudowires.

4.7. OWAMP and TWAMP

4.7.1. Overview

The IPPM working group in the IETF defines common criteria and metrics for measuring performance of IP traffic ([IPPM-FW]). Some of the key RFCs published by this working group have defined metrics for measuring connectivity [IPPM-Con], delay ([IPPM-1DM], [IPPM-2DM]), and packet loss [IPPM-1LM]. It should be noted that the work of the IETF in the context of performance metrics is not limited to IP networks; [PM-CONS] presents general guidelines for considering new performance metrics.

The IPPM working group has defined not only metrics for performance measurement, but also protocols that define how the measurement is carried out. The One-way Active Measurement Protocol [OWAMP] and the Two-Way Active Measurement Protocol [TWAMP] define a method and protocol for measuring performance metrics in IP networks.

OWAMP [OWAMP] enables measurement of one-way characteristics of IP networks, such as one-way packet loss and one-way delay. For its proper operation OWAMP requires accurate time of day setting at its end points.

TWAMP [TWAMP] is a similar protocol that enables measurement of both one-way and two-way (round trip) characteristics.

OWAMP and TWAMP are both comprised of two separate protocols:

- OWAMP-Control/TWAMP-Control: used to initiate, start, and stop test sessions and to fetch their results. Continuity Check and Connectivity Verification are tested and confirmed by establishing the OWAMP/TWAMP Control Protocol TCP connection.

- OWAMP-Test/TWAMP-Test: used to exchange test packets between two measurement nodes. Enables the loss and delay measurement functions, as well as detection of other anomalies, such as packet duplication and packet reordering.

It should be noted that while [OWAMP] and [TWAMP] define tools for performance measurement, they do not define the accuracy of these tools. The accuracy depends on scale, implementation and network configurations.

Alternative protocols for performance monitoring are defined, for example, in MPLS-TP OAM ([MPLS-LM-DM], [TP-LM-DM]), and in Ethernet OAM [ITU-T-Y1731].

4.7.2. Control and Test Protocols

OWAMP and TWAMP control protocols run over TCP, while the test protocols run over UDP. The purpose of the control protocols is to initiate, start, and stop test sessions, and for OWAMP to fetch results. The test protocols introduce test packets (which contain sequence numbers and timestamps) along the IP path under test according to a schedule, and record statistics of packet arrival. Multiple sessions may be simultaneously defined, each with a session identifier, and defining the number of packets to be sent, the amount of padding to be added (and thus the packet size), the start time, and the send schedule (which can be either a constant time between test packets or exponentially distributed pseudo-random). Statistics recorded conform to the relevant IPPM RFCs.

From a security perspective, OWAMP and TWAMP test packets are hard to detect because they are simply UDP streams between negotiated port numbers, with potentially nothing static in the packets. OWAMP and TWAMP also include optional authentication and encryption for both control and test packets.

4.7.3. OWAMP

OWAMP defines the following logical roles: Session-Sender, Session-Receiver, Server, Control-Client, and Fetch-Client. The Session-Sender originates test traffic that is received by the Session-Receiver. The Server configures and manages the session, as well as returning the results. The Control-Client initiates requests for test sessions, triggers their start, and may trigger their termination. The Fetch-Client requests the results of a completed session. Multiple roles may be combined in a single host - for example, one host may play the roles of Control-Client, Fetch-Client, and Session-Sender, and a second playing the roles of Server and Session-Receiver.

In a typical OWAMP session the Control-Client establishes a TCP connection to port 861 of the Server, which responds with a server greeting message indicating supported security/integrity modes. The Control-Client responds with the chosen communications mode and the Server accepts the mode. The Control-Client then requests and fully describes a test session to which the Server responds with its acceptance and supporting information. More than one test session may be requested with additional messages. The Control-Client then starts a test session and the Server acknowledges, and instructs the Session-Sender to start the test. The Session-Sender then sends test packets with pseudorandom padding to the Session-Receiver until the session is complete or until the Control-client stops the session. Once finished, the Session-Sender reports to the Server which recovers data from the Session-Receiver. The Fetch-Client can then send a fetch request to the Server, which responds with an acknowledgement and immediately thereafter the result data.

4.7.4. TWAMP

TWAMP defines the following logical roles: session-sender, session-reflector, server, and control-client. These are similar to the OWAMP roles, except that the Session-Reflector does not collect any packet information, and there is no need for a Fetch-Client.

In a typical TWAMP session the Control-Client establishes a TCP connection to port 862 of the Server, and mode is negotiated as in OWAMP. The Control-Client then requests sessions and starts them. The Session-Sender sends test packets with pseudorandom padding to the Session-Reflector which returns them with insertion of timestamps.

4.8. TRILL

The requirements of OAM in TRILL are defined in [TRILL-OAM]. The challenge in TRILL OAM, much like in MPLS networks, is that traffic between RBridges RB1 and RB2 may be forwarded through more than one path. Thus, an OAM protocol between RBridges RB1 and RB2 must be able to monitor all the available paths between the two RBridge.

During the writing of this document the detailed definition of the TRILL OAM tools are still work in progress. This subsection presents the main requirements of TRILL OAM.

The main requirements defined in [TRILL-OAM] are:

- Continuity Checking (CC) - the TRILL OAM protocol must support a function for CC between any two RBridges RB1 and RB2.
- Connectivity Verification (CV) - connectivity between two RBridges RB1 and RB2 can be verified on a per-flow basis.
- Path Tracing - allows an RBridge to trace all the available paths to a peer RBridge.
- Performance monitoring - allows an RBridge to monitor the packet loss and packet delay to a peer RBridge.

5. Summary

This section summarizes the OAM tools and functions presented in this document. This summary is an index to some of the main OAM tools defined in the IETF. This compact index that can be useful to all readers from network operators to standards development organizations. The summary includes a short subsection that presents some guidance to network equipment vendors.

5.1. Summary of OAM Tools

This subsection provides a short summary of each of the OAM toolsets described in this document.

A detailed list of the RFCs related to each toolset is given in Appendix A.1.

Toolset	Description	Transport Technology

IP Ping	Ping ([IntHost], [NetTerms]) is a simple application for testing reachability that uses ICMP Echo messages ([ICMPv4], [ICMPv6]).	IPv4/IPv6
IP Traceroute	Traceroute ([TCPIP-Tools], [NetTools]) is an application that allows users to trace the path between an IP source and an IP destination, i.e., to identify the nodes along the path. If more than one path exists between the source and destination Traceroute traces *a* path. The most common implementation of Traceroute uses UDP probe messages, although there are other implementations that use different probes, such as ICMP or TCP. Paris Traceroute [PARIS] is an extension that attempts to discovers all the available paths from A to B by scanning different values of header fields.	IPv4/IPv6
BFD	Bidirectional Forwarding Detection (BFD) is defined in [BFD] as a framework for a lightweight generic OAM tool. The intention is to define a base tool that can be used with various encapsulation types, network environments, and in various medium types.	generic
MPLS OAM	MPLS LSP Ping, as defined in [MPLS-OAM], [MPLS-OAM-FW] and [LSP-Ping], is an OAM tool for point-to-point and point-to-multipoint MLPS LSPs. It includes two main functions: Ping and Traceroute. BFD [BFD-LSP] is an alternative means for detecting MPLS LSP data plane failures.	MPLS
MPLS-TP OAM	MPLS-TP OAM is defined in a set of RFCs.	MPLS-TP

	The OAM requirements for MPLS Transport Profile (MPLS-TP) are defined in [MPLS-TP-OAM]. Each of the tools in the OAM toolset is defined in its own RFC, as specified in Section A.1.	
Pseudowire OAM	The PWE3 OAM architecture defines control channels that support the use of existing IETF OAM tools to be used for a pseudo-wire (PW). The control channels that are defined in [VCCV] and [PW-G-ACh] may be used in conjunction with ICMP Ping, LSP Ping, and BFD to perform CC and CV functionality. In addition the channels support use of any of the MPLS-TP based OAM tools for completing their respective OAM functionality for a PW.	Pseudowire
OWAMP and TWAMP	The One Way Active Measurement Protocol [OWAMP] and the Two Way Active Measure-ment Protocols [TWAMP] are two protocols defined in the IP Performance Metrics (IPPM) working group in the IETF. These protocols allow various performance metrics to be measured, such as packet loss, delay and delay variation, duplication and reordering.	IPv4/IPv6
TRILL OAM	The requirements of OAM in TRILL are defined in [TRILL-OAM]. These requirements include continuity checking, connectivity verification, path tracing and performance monitoring. During the writing of this document the detailed definition of the TRILL OAM tools is work in progress.	TRILL

Table 3 Summary of OAM-related IETF Tools

5.2. Summary of OAM Functions

Table 4 summarizes the OAM functions that are supported in each of the toolsets that were analyzed in this section. The columns of this tables are the typical OAM functions described in Section 1.3.

Toolset	Continuity Check	Connectivity Verification	Path Discovery	Performance Monitoring	Other Functions
IP Ping	Echo				
IP Traceroute			Traceroute		
BFD	BFD Control/ Echo	BFD Control			RDI using BFD Control
MPLS OAM (LSP Ping)		"Ping" mode	"Traceroute" mode		
MPLS-TP OAM	CC	CV/pro-active or on-demand	Route Tracing	-LM -DM	-Diagnostic Test -Lock -Alarm Reporting -Client Failure Indication -RDI
Pseudowire OAM	BFD	-BFD -ICMP Ping -LSP-Ping	LSP-Ping		
OWAMP and	- control			-Delay	

TWAMP protocol			measurement-Packet loss measurement		
TRILL OAM	CC	CV	Path tracing	Delay measurement-Packet loss measurement	

Table 4 Summary of the OAM Functionality in IETF OAM Tools

5.3. Guidance to Network Equipment Vendors

As mentioned in Section 1.4. , it is imperative for OAM tools to be capable of testing the actual data plane in as much accuracy as possible. While this guideline may appear obvious, it is worthwhile to emphasize the key importance of enforcing fate-sharing between OAM traffic that monitors the data plane and the data plane traffic it monitors.

6. Security Considerations

OAM is tightly coupled with the stability of the network. A successful attack on an OAM protocol can create a false illusion of non-existent failures, or prevent the detection of actual ones. In both cases the attack may result in denial of service.

Some of the OAM tools presented in this document include security mechanisms that provide integrity protection, thereby preventing attackers from forging or tampering with OAM packets. For example, [BFD] includes an optional authentication mechanism for BFD Control packets, using either SHA1, MD5, or a simple password. [OWAMP] and [TWAMP] have 3 modes of security: unauthenticated, authenticated, and encrypted. The authentication uses SHA1 as the HMAC algorithm, and the encrypted mode uses AES encryption.

Confidentiality is typically not considered a requirement for OAM protocols. However, the use of encryption (e.g., [OWAMP] and

[TWAMP]) can make it difficult for attackers to identify OAM packets, thus making it more difficult to attack the OAM protocol.

OAM can also be used as a means for network reconnaissance; information about addresses, port numbers and about the network topology and performance can be gathered either by passively eavesdropping to OAM packets, or by actively sending OAM packets and gathering information from the respective responses. This information can then be used maliciously to attack the network. Note that some of this information, e.g., addresses and port numbers, can be gather even when encryption is used ([OWAMP], [TWAMP]).

For further details about the security considerations of each OAM protocol, the reader is encouraged to review the Security Considerations section of each document referenced by this memo.

7. IANA Considerations

There are no new IANA considerations implied by this document.

8. Acknowledgments

The authors gratefully acknowledge Sasha Vainshtein, Carlos Pignataro, David Harrington, Dan Romascanu, Ron Bonica, Benoit Claise, Stewart Bryant, Tom Nadeau, Elwyn Davies, Al Morton, Sam Aldrin, Thomas Narten, and other members of the OPSA WG for their helpful comments on the mailing list.

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

9. References

9.1. Normative References

[OAM-Def] Andersson, L., Van Helvoort, H., Bonica, R., Romascanu, D., Mansfield, S., "Guidelines for the use of the OAM acronym in the IETF ", RFC 6291, June 2011.

9.2. Informative References

[ATM-L2] Singh, S., Townsley, M., and C. Pignataro, "Asynchronous Transfer Mode (ATM) over Layer 2 Tunneling Protocol Version 3 (L2TPv3)", RFC 4454, May 2006.

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

[BFD-Gen] Katz, D., Ward, D., "Generic Application of Bidirectional Forwarding Detection (BFD)", RFC 5882, June 2010.

[BFD-IP] Katz, D., Ward, D., "Bidirectional Forwarding Detection (BFD) for IPv4 and IPv6 (Single Hop)", RFC 5881, June 2010.

[BFD-LSP] Aggarwal, R., Kompella, K., Nadeau, T., and Swallow, G., "Bidirectional Forwarding Detection (BFD) for MPLS Label Switched Paths (LSPs)", RFC 5884, June 2010.

[BFD-Multi] Katz, D., Ward, D., "Bidirectional Forwarding Detection (BFD) for Multihop Paths", RFC 5883, June 2010.

[BFD-VCCV] Nadeau, T., Pignataro, C., "Bidirectional Forwarding Detection (BFD) for the Pseudowire Virtual Circuit Connectivity Verification (VCCV)", RFC 5885, June 2010.

[Comp] Bonaventure, O., "Computer Networking: Principles, Protocols and Practice", 2008.

[Dup] Uijterwaal, H., "A One-Way Packet Duplication Metric", RFC 5560, May 2009.

[Eth-Int] Mohan, D., Bitar, N., Sajassi, A., Delord, S., Niger, P., Qiu, R., "MPLS and Ethernet Operations, Administration, and Maintenance (OAM) Interworking", RFC 7023, October 2013.

[G-ACh] Bocci, M., Vigoureux, M., Bryant, S., "MPLS Generic Associated Channel", RFC 5586, June 2009.

[ICMP-Ext] Bonica, R., Gan, D., Tappan, D., Pignataro, C., "ICMP Extensions for Multiprotocol Label Switching", RFC 4950, August 2007.

[ICMP-Int] Atlas, A., Bonica, R., Pignataro, C., Shen, N., Rivers, JR., "Extending ICMP for Interface and Next-Hop Identification", RFC 5837, April 2010.

[ICMP-MP] Bonica, R., Gan, D., Tappan, D., Pignataro, C., "Extended ICMP to Support Multi-Part Messages", RFC 4884, April 2007.

[ICMPv4] Postel, J., "Internet Control Message Protocol", STD 5, RFC 792, September 1981.

[ICMPv6] Conta, A., Deering, S., and M. Gupta, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", RFC 4443, March 2006.

[IEEE802.1Q] IEEE 802.1Q, "IEEE Standard for Local and metropolitan area networks - Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks", October 2012.

[IEEE802.3ah] IEEE 802.3, "IEEE Standard for Information technology - Local and metropolitan area networks - Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications", clause 57, December 2008.

[IntHost] Braden, R., "Requirements for Internet Hosts -- Communication Layers", RFC 1122, October 1989.

[IPPM-1DM] Almes, G., Kalidindi, S., Zekauskas, M., "A One-way Delay Metric for IPPM", RFC 2679, September 1999.

[IPPM-1LM] Almes, G., Kalidindi, S., Zekauskas, M., "A One-way Packet Loss Metric for IPPM", RFC 2680, September 1999.

[IPPM-2DM] Almes, G., Kalidindi, S., Zekauskas, M., "A Round-trip Delay Metric for IPPM", RFC 2681, September 1999.

[IPPM-Con] Mahdavi, J., Paxson, V., "IPPM Metrics for Measuring Connectivity", RFC 2678, September 1999.

[IPPM-FW] Paxson, V., Almes, G., Mahdavi, J., and Mathis, M., "Framework for IP Performance Metrics", RFC 2330, May 1998.

[ITU-G8113.1] ITU-T Recommendation G.8113.1/Y.1372.1, "Operations, Administration and Maintenance mechanism for MPLS-TP in Packet Transport Network (PTN)", November 2012.

[ITU-G8113.2] ITU-T Recommendation G.8113.2/Y.1372.2, "Operations, administration and maintenance mechanisms for MPLS-TP networks using the tools defined for MPLS", November 2012.

[ITU-T-CT] Betts, M., "Allocation of a Generic Associated Channel Type for ITU-T MPLS Transport Profile Operation, Maintenance, and Administration (MPLS-TP OAM)", RFC 6671, November 2012.

[ITU-T-G.806] ITU-T Recommendation G.806, "Characteristics of transport equipment - Description methodology and generic functionality", January 2009.

[ITU-T-Y1711] ITU-T Recommendation Y.1711, "Operation & Maintenance mechanism for MPLS networks", February 2004.

[ITU-T-Y1731] ITU-T Recommendation G.8013/Y.1731, "OAM Functions and Mechanisms for Ethernet-based Networks", July 2011.

[ITU-Terms] ITU-R/ITU-T Terms and Definitions, online, 2013.

[L2TP-EC] McGill, N. and C. Pignataro, "Layer 2 Tunneling Protocol Version 3 (L2TPv3) Extended Circuit Status Values", RFC 5641, August 2009.

[L2VPN-OAM] Sajassi, A., Mohan, D., "Layer 2 Virtual Private Network (L2VPN) Operations, Administration, and Maintenance (OAM) Requirements and Framework", RFC 6136, March 2011.

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

[Lock-Loop] Boutros, S., Sivabalan, S., Aggarwal, R., Vigoureux, M., Dai, X., "MPLS Transport Profile Lock Instruct and Loopback Functions", RFC 6435, November 2011.

[LSP-Ping] Kompella, K., Swallow, G., "Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures", RFC 4379, February 2006.

[Mng] Farrel, A., "Inclusion of Manageability Sections in Path Computation Element (PCE) Working Group Drafts", RFC 6123, February 2011.

[MPLS-ENCAPS] Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y., Farinacci, D., Li, T. and A. Conta, "MPLS Label Stack Encoding", RFC 3032, January 2001.

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

[MPLS-OAM] Nadeau, T., Morrow, M., Swallow, G., Allan, D., Matsushima, S., "Operations and Management (OAM) Requirements for Multi-Protocol Label Switched (MPLS) Networks", RFC 4377, February 2006.

[MPLS-OAM-FW] Allan, D., Nadeau, T., "A Framework for Multi-Protocol Label Switching (MPLS) Operations and Management (OAM)", RFC 4378, February 2006.

[MPLS-P2MP] Yasukawa, S., Farrel, A., King, D., Nadeau, T., "Operations and Management (OAM) Requirements for Point-to-Multipoint MPLS Networks", RFC 4687, September 2006.

[MPLS-TP-OAM] Vigoureux, M., Ward, D., Betts, M., "Requirements for OAM in MPLS Transport Networks", RFC 5860, May 2010.

[mtrace] Fenner, W., Casner, S., "A "traceroute" facility for IP Multicast", draft-ietf-idmr-traceroute-ipm-07 (expired), July 2000.

[NetTerms] Jacobsen, O., Lynch, D., "A Glossary of Networking Terms", RFC 1208, March 1991.

[NetTools] Enger, R., Reynolds, J., "FYI on a Network Management Tool Catalog: Tools for Monitoring and Debugging TCP/IP Internets and Interconnected Devices", RFC 1470, June 1993.

[OAM-Analys] Sprecher, N., Fang, L., "An Overview of the OAM Tool Set for MPLS based Transport Networks", RFC 6669, July 2012.

[OAM-Label] Ohta, H., "Assignment of the 'OAM Alert Label' for Multiprotocol Label Switching Architecture (MPLS) Operation and Maintenance (OAM) Functions", RFC 3429, November 2002.

[OAM-Mng] Ersue, M., Claise, B., "An Overview of the IETF Network Management Standards", RFC 6632, June 2012.

[OnDemand-CV] Gray, E., Bahadur, N., Boutros, S., Aggarwal, R. "MPLS On-Demand Connectivity Verification and Route Tracing", RFC 6426, November 2011.

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

[PARIS] Brice Augustin, Timur Friedman and Renata Teixeira, "Measuring Load-balanced Paths in the Internet", IMC, 2007.

[PM-CONS] Clark, A. and B. Claise, "Guidelines for Considering New Performance Metric Development", BCP 170, RFC 6390, October 2011.

[PW-ACH] Bryant, S., Swallow, G., Martini, L., McPherson, D., "Pseudowire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN", RFC 4385, February 2006.

[PW-G-ACh] Li, H., Martini, L., He, J., Huang, F., "Using the Generic Associated Channel Label for Pseudowire in the MPLS Transport Profile (MPLS-TP)", RFC 6423, November 2011.

[PW-MAP] Aissaoui, M., Busschbach, P., Martini, L., Morrow, M., Nadeau, T., and Y(J). Stein, "Pseudowire (PW) Operations, Administration, and Maintenance (OAM) Message Mapping", RFC 6310, July 2011.

[Reorder] Morton, A., Ciavattone, L., Ramachandran, G., Shalunov, S., and J. Perser, "Packet Reordering Metrics", RFC 4737, November 2006.

[Signal] Yasukawa, S., "Signaling Requirements for Point-to-Multipoint Traffic-Engineered MPLS Label Switched Paths (LSPs)", RFC 4461, April 2006.

[TCPIP-Tools] Kessler, G., Shepard, S., "A Primer On Internet and TCP/IP Tools and Utilities", RFC 2151, June 1997.

[TP-CC-CV] Allan, D., Swallow, G., Drake, J., "Proactive Connectivity Verification, Continuity Check and Remote Defect indication for MPLS Transport Profile", RFC 6428, November 2011.

[TP-Fault] Swallow, G., Fulignoli, A., Vigoureux, M., Boutros, S., "MPLS Fault Management Operations, Administration, and Maintenance (OAM)", RFC 6427, November 2011.

[TP-LM-DM] Frost, D., Bryant, S., "A Packet Loss and Delay Measurement Profile for MPLS-Based Transport Networks", RFC 6375, September 2011.

[TP-OAM-FW] Busi, I., Allan, D., "Operations, Administration and Maintenance Framework for MPLS-based Transport Networks ", RFC 6371, September 2011.

[TP-Term] Van Helvoort, H., Andersson, L., Sprecher, N., "A Thesaurus for the Terminology used in MPLS Transport Profile (MPLS-TP) Internet-Drafts and RFCs in the Context of the ITU-T's Transport Network Recommendations", RFC 7087, December 2013.

[TRILL-OAM] Senevirathne, T., Bond, D., Aldrin, S., Li, Y., Watve, R., "Requirements for Operations, Administration, and Maintenance (OAM) in Transparent Interconnection of Lots of Links (TRILL)", RFC 6905, March 2013.

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

[VCCV] Nadeau, T., Pignataro, C., "Pseudowire Virtual Circuit Connectivity Verification (VCCV): A Control Channel for Pseudowires", RFC 5085, December 2007.

[VCCV-SURVEY] Del Regno, N., Malis, A., "The Pseudowire (PW) and Virtual Circuit Connectivity Verification (VCCV) Implementation Survey Results", RFC 7079, November 2013.

Appendix A. List of OAM Documents

A.1. List of IETF OAM Documents

Table 5 summarizes the OAM related RFCs published by the IETF.

It is important to note that the table lists various RFCs that are different by nature. For example, some of these documents define OAM tools or OAM protocols (or both), while others define protocols that

are not strictly OAM-related, but are used by OAM tools. The table also includes RFCs that define the requirements or the framework of OAM in a specific context (e.g., MPLS-TP).

The RFCs in the table are categorized in a few sets as defined in Section 1.3.

Toolset	Title	RFC
IP Ping	Requirements for Internet Hosts -- Communication Layers [IntHost]	RFC 1122
	A Glossary of Networking Terms [NetTerms]	RFC 1208
	Internet Control Message Protocol [ICMPv4]	RFC 792
	Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification [ICMPv6]	RFC 4443
IP Traceroute	A Primer On Internet and TCP/IP Tools and Utilities [TCPIP-Tools]	RFC 2151
	FYI on a Network Management Tool Catalog: Tools for Monitoring and Debugging TCP/IP Internets and Interconnected Devices [NetTools]	RFC 1470
	Internet Control Message Protocol [ICMPv4]	RFC 792
	Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification [ICMPv6]	RFC 4443
	Extended ICMP to Support Multi-Part Messages [ICMP-MP]	RFC 4884

	Extending ICMP for Interface and Next-Hop Identification [ICMP-Int]	RFC 5837
BFD	Bidirectional Forwarding Detection [BFD]	RFC 5880
	Bidirectional Forwarding Detection (BFD) for IPv4 and IPv6 (Single Hop) [BFD-IP]	RFC 5881
	Generic Application of Bidirectional Forwarding Detection [BFD-Gen]	RFC 5882
	Bidirectional Forwarding Detection (BFD) for Multihop Paths [BFD-Multi]	RFC 5883
	Bidirectional Forwarding Detection for MPLS Label Switched Paths (LSPs) [BFD-LSP]	RFC 5884
	Bidirectional Forwarding Detection for the Pseudowire Virtual Circuit Connectivity Verification (VCCV) [BFD-VCCV]	RFC 5885
MPLS OAM	Operations and Management (OAM) Requirements for Multi-Protocol Label Switched (MPLS) Networks [MPLS-OAM]	RFC 4377
	A Framework for Multi-Protocol Label Switching (MPLS) Operations and Management (OAM) [MPLS-OAM-FW]	RFC 4378
	Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures [LSP-Ping]	RFC 4379
	Operations and Management (OAM) Requirements for Point-to-Multipoint MPLS Networks [MPLS-P2MP]	RFC 4687

		ICMP Extensions for Multiprotocol Label Switching [ICMP-Ext]	RFC 4950
		Bidirectional Forwarding Detection for MPLS Label Switched Paths (LSPs) [BFD-LSP]	RFC 5884
MPLS-TP OAM		Requirements for OAM in MPLS-TP [MPLS-TP-OAM]	RFC 5860
		MPLS Generic Associated Channel [G-ACh]	RFC 5586
		MPLS-TP OAM Framework [TP-OAM-FW]	RFC 6371
		Proactive Connectivity Verification, Continuity Check, and Remote Defect Indication for the MPLS Transport Profile [TP-CC-CV]	RFC 6428
		MPLS On-Demand Connectivity Verification and Route Tracing [OnDemand-CV]	RFC 6426
		MPLS Fault Management Operations, Administration, and Maintenance (OAM) [TP-Fault]	RFC 6427
		MPLS Transport Profile Lock Instruct and Loopback Functions [Lock-Loop]	RFC 6435
		Packet Loss and Delay Measurement for MPLS Networks [MPLS-LM-DM]	RFC 6374
		A Packet Loss and Delay Measurement Profile for MPLS-Based Transport Networks [TP-LM-DM]	RFC 6375
Pseudowire		Pseudowire Virtual Circuit	RFC 5085

OAM	Connectivity Verification (VCCV): A Control Channel for Pseudowires [VCCV]	
	Bidirectional Forwarding Detection for the Pseudowire Virtual Circuit Connectivity Verification (VCCV) [BFD-VCCV]	RFC 5885
	Using the Generic Associated Channel Label for Pseudowire in the MPLS Transport Profile (MPLS-TP) [PW-G-ACh]	RFC 6423
	Pseudowire (PW) Operations, Administration, and Maintenance (OAM) Message Mapping [PW-MAP]	RFC 6310
	MPLS and Ethernet Operations, Administration, and Maintenance (OAM) Interworking [Eth-Int]	RFC 7023
OWAMP and TWAMP	A One-way Active Measurement Protocol [OWAMP]	RFC 4656
	A Two-Way Active Measurement Protocol [TWAMP]	RFC 5357
	Framework for IP Performance Metrics [IPPM-FW]	RFC 2330
	IPPM Metrics for Measuring Connectivity [IPPM-Con]	RFC 2678
	A One-way Delay Metric for IPPM [IPPM-1DM]	RFC 2679
	A One-way Packet Loss Metric for IPPM [IPPM-1LM]	RFC 2680
	A Round-trip Delay Metric for IPPM	RFC 2681

	[IPPM-2DM]		
	Packet Reordering Metrics [Reorder]	RFC 4737	
	A One-Way Packet Duplication Metric [Dup]	RFC 5560	
TRILL OAM	Requirements for Operations, Administration, and Maintenance (OAM) in Transparent Interconnection of Lots of Links (TRILL)	RFC 6905	

Table 5 Summary of IETF OAM Related RFCs

A.2. List of Selected Non-IETF OAM Documents

In addition to the OAM tools defined by the IETF, the IEEE and ITU-T have also defined various OAM tools that focus on Ethernet, and various other transport network environments. These various tools, defined by the three standard organizations, are often tightly coupled, and have had a mutual effect on each other. The ITU-T and IETF have both defined OAM tools for MPLS LSPs, [ITU-T-Y1711] and [LSP-Ping]. The following OAM standards by the IEEE and ITU-T are to some extent linked to IETF OAM tools listed above and are mentioned here only as reference material:

- OAM tools for Layer 2 have been defined by the ITU-T in [ITU-T-Y1731], and by the IEEE in 802.1ag [IEEE802.1Q] . The IEEE 802.3 standard defines OAM for one-hop Ethernet links [IEEE802.3ah].

- The ITU-T has defined OAM for MPLS LSPs in [ITU-T-Y1711], and MPLS-TP OAM in [ITU-G8113.1] and [ITU-G8113.2].

It should be noted that these non-IETF documents deal in many cases with OAM functions below the IP layer (Layer 2, Layer 2.5) and in some cases operators use a multi-layered OAM approach, which is a function of the way their networks are designed.

Table 6 summarizes some of the main OAM standards published by non-IETF standard organizations. This document focuses on IETF OAM standards, but these non-IETF standards are referenced in this document where relevant.

	Title	Standard/Draft
ITU-T MPLS OAM	Operation & Maintenance mechanism for MPLS networks [ITU-T-Y1711]	ITU-T Y.1711
	Assignment of the 'OAM Alert Label' for Multiprotocol Label Switching Architecture (MPLS) Operation and Maintenance (OAM) Functions [OAM-Label] Note: although this is an IETF document, it is listed as one of the non-IETF OAM standards, since it was defined as a complementary part of ITU-T Y.1711.	RFC 3429
ITU-T MPLS-TP OAM	Operations, administration and Maintenance mechanisms for MPLS-TP networks using the tools defined for MPLS [ITU-G8113.2] Note: this document describes the OAM toolset defined by the IETF for MPLS-TP, whereas ITU-T G.8113.1 describes the OAM toolset defined by the ITU-T.	ITU-T G.8113.2
	Operations, Administration and Maintenance mechanism for MPLS-TP in Packet Transport Network (PTN)	ITU-T G.8113.1
	Allocation of a Generic Associated Channel Type for ITU-T MPLS Transport Profile Operation, Maintenance, and Administration (MPLS-TP OAM) [ITU-T-CT] Note: although this is an IETF document, it is listed as one of the	RFC 6671

	non-IETF OAM standards, since it was defined as a complementary part of ITU-T G.8113.1.		
ITU-T Ethernet OAM	OAM Functions and Mechanisms for Ethernet-based Networks [ITU-T-Y1731]	ITU-T Y.1731	
IEEE CFM	Connectivity Fault Management [IEEE802.1Q] Note: CFM was originally published as IEEE 802.1ag, but is now incorporated in the 802.1Q standard.	IEEE 802.1ag	
IEEE DDCFM	Management of Data Driven and Data Dependent Connectivity Faults [IEEE802.1Q] Note: DDCFM was originally published as IEEE 802.1Qaw, but is now incorporated in the 802.1Q standard.	IEEE 802.1ag	
IEEE 802.3 link level OAM	Media Access Control Parameters, Physical Layers, and Management Parameters for Subscriber Access Networks [IEEE802.3ah] Note: link level OAM was originally defined in IEEE 802.3ah, and is now incorporated in the 802.3 standard.	IEEE 802.3ah	

Table 6 Non-IETF OAM Standards Mentioned in this Document

Authors' Addresses

Tal Mizrahi
Marvell
6 Hamada St.
Yokneam, 20692
Israel

Email: talmi@marvell.com

Nurit Sprecher
Nokia Solutions and Networks
3 Hanagar St. Neve Ne'eman B
Hod Hasharon, 45241
Israel

Email: nurit.sprecher@nsn.com

Elisa Bellagamba
Ericsson
6 Farogatan St.
Stockholm, 164 40
Sweden

Phone: +46 761440785
Email: elisa.bellagamba@ericsson.com

Yaacov Weingarten
34 Hagefen St.
Karnei Shomron, 4485500
Israel

Email: wyaacov@gmail.com

Network Working Group
Internet-Draft
Intended status: Informational
Expires: January 17, 2013

H. Okita
M. Yoshizawa
T. Suzuki
T. Iijima
Hitachi, Ltd.
July 16, 2012

Virtual Network Management Information Model
draft-okita-ops-vnetmodel-07

Abstract

Virtual switches on server virtualization platforms cause a problem in managing networks in data center and between data centers containing several hundred switches. Accordingly, a management information model for the networks containing virtual switches is proposed. The proposed model consists of a physical layer (which represents connections between physical switches) and a virtual layer (which represents connections between virtual switches). These layers also represent the association of the virtual switch with the corresponding physical switch. This document also provides implementation examples of proposed information model in XML and Yang.

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 http://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 17, 2013.

Copyright Notice

Table of Contents

1. Introduction

In data center networks, a virtual switch on a server virtualization platform works as a virtual network element [VEB] [EVB-PAR] [PE-PAR] . The virtual switch connects multiple virtual machines on the same server virtualization platform and connects these virtual machines to external physical switches.

Virtual switches, however, cause a problem in managing data center networks because, mainly, a virtual switch and a physical switch require different management systems. Operators of networks therefore have to use multiple management systems for managing the whole network.

To avoid this management difficulty, an integrated network management system (NMS) is effective. The integrated NMS collects and stores virtual-network management information that describes network structure of a managed target network. It then displays or transmits this management information as a response to a request from operators or other NMSs.

The purpose of this document is to provide a management information model that represents the network structure of the whole network including data centers containing virtual switches. Section 2 describes the problem statement, Section 3 describes the necessity of a virtual network management system, Section 4 describes power saving use case, Section 5 describes requrements for the information model, Section 6 describes the relationships to the existing MIBs, Section 7 defines the propsed information model, Section 8 describes an XML Schema based data model of the information model, Section 9 describes a Yang module of the information model.

2. Problem Statement

Virtual switches cause a difficulty in managing networks including data centers. They expand the data center network into the server virtualization platforms. Therefore, to manage the whole networks, network operators have to manage virtual switches in addition to physical switches.

To manage these virtual and physical switches, the operators have to use multiple management interfaces. Specifically, to manage virtual switches, they have to use a specific management system for the server virtualization platform that the target virtual switches are created on. Moreover, to manage physical switches, they use a network management system. Figure 1 shows an architectural overview of a conventional network management system.

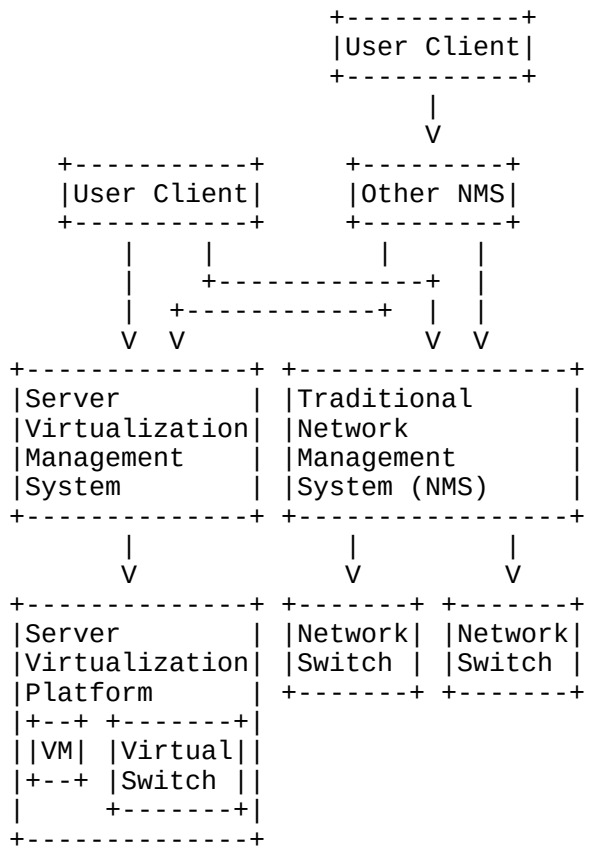

Figure 1: Overview of a network management system

This conventional management architecture causes the following problems which increase the operation time taken by operators of the networks and thus increase operational costs.

1. When operators want to examine the network structure of a virtual network containing virtual switches, they have to access multiple management systems.

2. When operators want to examine the mapping of a virtual network to corresponding physical components, they have to access multiple management systems.

3. When operators want to configure a data center network according to a VM migration in the data center, they have to access multiple management systems.

4. When operators want to configure a network among data centers according to a VM migration over the data centers, they have to access multiple management systems.

5. When operators want to configure multi-layer networks for a power-saving cloud system consisting of multiple data centers and networks, they have to access multiple management systems.

To solve these problems and save the operation time for the networks, the following requirements must be met.

1. The data center network should provide an integrated management system that enables operators to get network structure information about virtual network.

2. The data center network should provide an integrated management system that enables operators to get mapping information about virtual switches and their underlying physical platforms.

3. The data center network should provide an integrated management system that enables operators to configure the data center network including virtual switches.

4. The network including data centers should provide an integrated management system that enables operators to configure the whole network including virtual networks.

5. The network including data centers should provide an integrated management system that enables operators to configure multi-layer networks including not only physical networks but also virtual

networks.

3. Virtual Network Management System

A system architecture that effectively satisfies the above-described requirements is proposed in the following.

An integrated network management system (NMS) effectively reduces the network operation time needed for managing virtual switches and physical switches. It is referred to as a VNMS (Virtual Network Management System.) It integrates multiple existing management interfaces into a single interface. Operators can thus reduce their operation time.

The VNMS manages device connectivity in the managed target network. To perform this task, it stores network management information about configured virtual networks in the target network.

Figure 2 shows an overview of the system architecture of the target system. The virtual-network management information about the VNMS is based on the proposed model .

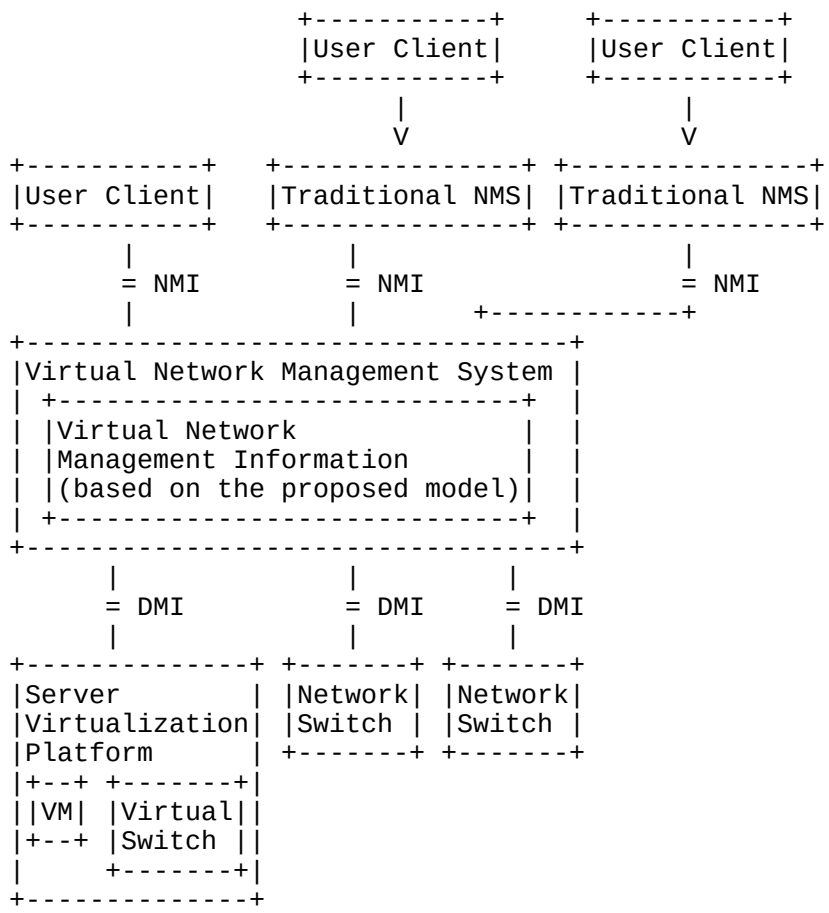

Figure 2: Overview of system architecture

The following three types of elements exist around this VNMS.

- User clients or traditional NMS
- Network switches
- Server virtualization platforms

The user client or network application uses management information about device connections in the managed network. The network switches are virtualized as multiple virtual switches. Moreover, the server virtualization platforms are virtualized as multiple virtual machines and internal virtual switches. A set of virtual switches

and virtual machines forms a virtual system for a user.

Among the elements described above, we define the following two management interfaces.

- Network Management Interface (NMI)
- Device Management Interface (DMI)

The network management interface (NMI) is set between the network application and the VNMS. This interface is used by the VNMS to transport virtual-network management information to network applications in response to their request.

Datamodels provide the definition and format of the virtual-network management information transported on the NMI. The definition describes an encoding scheme and an underlying transport protocol. The VNMS may use, for example, SNMP (Simple Network Management Protocol) and MIB (Management Information Base) specified in the Internet-standard management framework[RFC3410] or an XML-based management framework[RFC3535] as the datamodel.

The device-management interface (DMI) is set between the VNMS and network devices, which include the server virtualization platforms and network switches. The DMI is used by the VNMS to query management information about a target device. This interface is device specific and not standardized by this document.

4. Power Saving Use Case

One of use cases for the virtual network managament system (VNMS) is an optimization of the VMs' location for power saving as shown in Figure 3. The example system is composed of the VNMS, multiple data centers consisting of many servers, an inter data center network, and a network application.

When the network application optimizes the VMs' location, one or more VMs are needed to be relocated between the servers in the multiple data centers. To execute that, the network application needs to know the current situations of the whole network including the data center networks. In addition, it needs to configure the network to meet conditions for the relocated VMs.

In this case, if there is no common interface for the network management, the network application needs to use many traditional interfaces for the configuration of networks. However, in the proposed model, it needs to only access the VNMS for getting the current network configurations and for managing the whole network including the data center networks through the NMI.

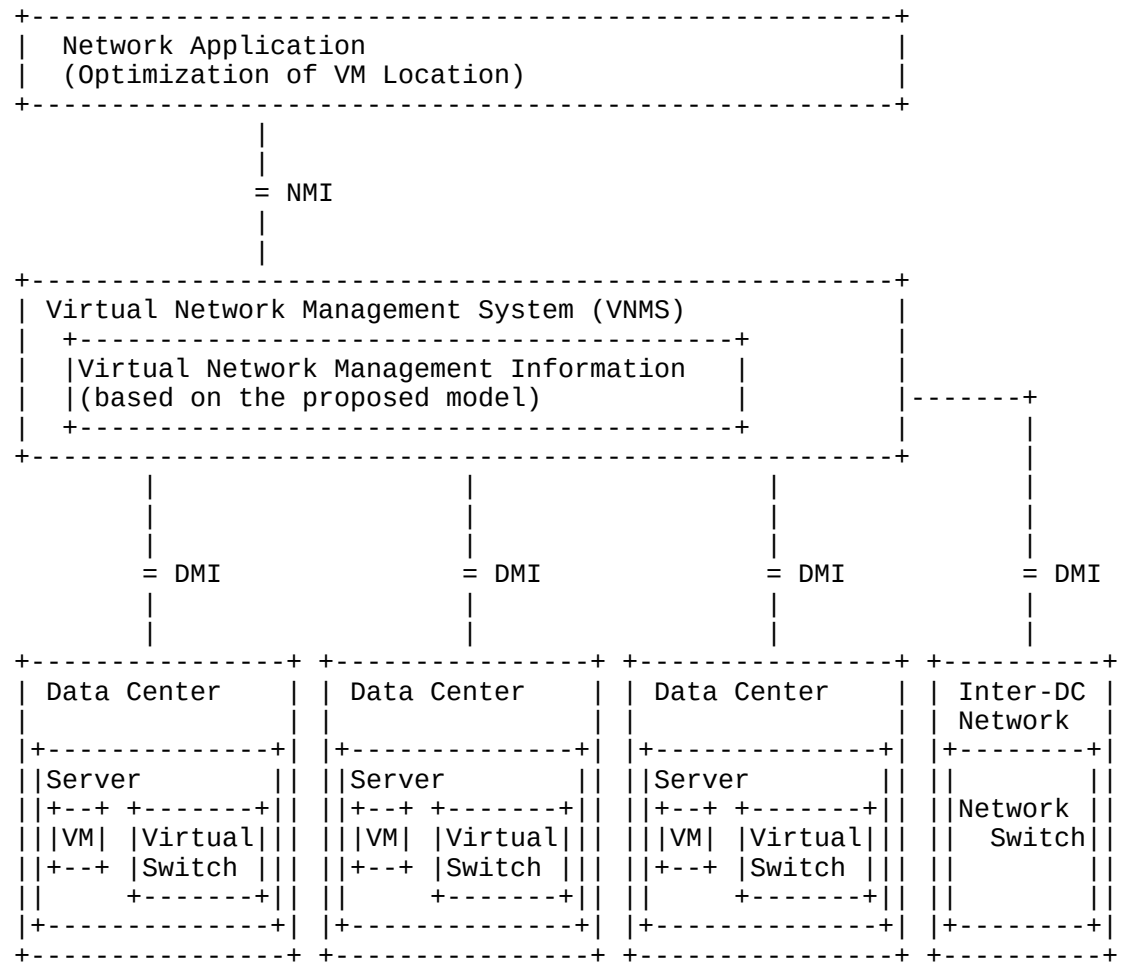

Figure 3: VM Location Optimization for Power Saving

5. Requirements for Virtual Network Management Information Model

This document focuses on an information model for the virtual-network management information described in the section 3. The requirements for the information model are listed below. These requirements arise from the problems stated in the previous sections.

1. Physical Resource Information: The proposed model should be able to represent the physical resources available on the target network. Those resources include several physical network devices, for example, network switches, routers. And, they also include server virtualization platforms.

2. Physical Hierarchy Information: The proposed model should be able to represent the hierarchy of physical resources in the target network. For example, the relationship between a chassis of a network switch and its network interface cards should be represented.

3. Physical Connection Information: The proposed model should represent a connection among physical switches and physical servers in the target network.

4. Virtual Resource Information: The proposed model should be able to represent the virtual resources available on the target network. Those resources include several virtual devices, for example, virtualized switches and virtual switches on the server virtualization platforms. And, they also include virtual machines on server virtualization platforms.

5. Virtual Connection Information: The proposed model should represent a connection between virtual switches and virtual machines in the target network.

6. Virtual-Physical Mapping Information: The proposed model should represent mapping of a virtual switch to the physical server that the virtual switch is created on.

6. Relationships to Existing MIBs

A lot of RFCs about MIBs have been published from the IETF. These existing MIBs provide each information models implicitly. For avoiding inventing the wheel, we researched relationships between the requirements for the virtual network management information model and existing MIBs.

6.1. Relationships to LLDP-MIB

Protocols for network topology discovery like Link Layer Discovery Protocol (LLDP) use some of MIB modules. These MIB modules are used to describe link state information in the managed network. For example, the LLDP-MIB [IEEE.802-1AB.2005] standardized as IEEE Standard 802.1AB supports this function.

The LLDP-MIB can be used to describe a connection between neighboring layer-2 MAC bridges. In the LLDP-MIB, there is an lldpRemTable which contains one or more rows per physical network connection. The row contains a chassis ID, a port ID, a port description, and system information for each neighboring layer-2 MAC bridge.

As described above, the LLDP-MIB can be used to describe the connection information between physical entities like physical switches. However, the LLDP-MIB cannot be used to describe the connection information between logical entities. Thus, it cannot be used to describe the connection information between a virtual switch and a virtual machine on the same physical server. Moreover, it cannot be used to describe the connection information between a virtual switch and an external physical switch.

As the result, the LLDP-MIB does not satisfy the first requirement in section 2 for the virtual network management information model.

6.2. Relationships to ENTITY-MIB

The ENTITY-MIB [RFC2737] was published by the IETF entmib WG. It can be used to represent a single SNMP agent which supports multiple instances of one MIB. For example, a single physical switch having a single SNMP agent can support multiple instances of a bridge with the ENTITY-MIB.

The ENTITY-MIB can be used to describe following two types of information.

One is mapping information between logical entities and physical entities on one network element. The information can be represented by the entLPMappingTable and the entAliasMappingTable in the

entityMapping group. For example, these tables support logical entities which contain OSPF instances and 802.1d bridges. Moreover, these tables support physical entities which contain bridge ports, backplanes and chassis.

Another is information about hierarchy relationship among physical entities. The information can be represented by the entPhysicalContainsTable in the entityMapping group. The entPhysicalContainsTable contains simple mapping information between 'container' entity and 'containee' entity. For example, a chassis is a 'container' entity. Its bridge ports and its backplane are 'containee' entities.

As described above, the ENTITY-MIB can be used to describe the mapping information between logical entities and physical entities. Therefore, the ENTITY-MIB satisfies the second requirement in section 2 for the virtual network management information model.

However, the ENTITY-MIB cannot be used to describe the connection information between logical entities. For example, it is impossible to describe connection information between virtual switches with the ENTITY-MIB.

As the result, the ENTITY-MIB does not satisfy the first requirement in section 2 for the virtual network management information model.

7. Proposals of Virtual Network Management Information Model

This section defines the proposed virtual-network management information model, which is an object-oriented information model. The model can satisfy both of the requirements included in section 2. The model is an abstract-information model independent from encoding schemes and management protocols. The model is written in Unified Modeling Language (UML) [UML] .

7.1. TargetedNetwork Object

The proposed model starts with a TargetedNetwork object. This object represents the overall network. In the network, two types of network exist: a physical network and a virtual network. In the proposed model, a PhysicalNetwork object represents a physical network, and a VirtualNetwork object represents a virtual network. To represent this structure, the TargetedNetwork object has one or multiple references to PhysicalNetwork objects and VirtualNetwork objects.

Furthermore, the PhysicalNetwork object and the VirtualNetwork have a reference between them. Since a physical network can create multiple virtual networks, the PhysicalNetwork object can have multiple references to corresponding VirtualNetwork objects. On the contrary, the VirtualNetwork object has only one reference to the PhysicalNetwork object, since the virtual network is created on the specific physical network.

Figure 4 shows a class diagram of the proposed virtual-network management information model containing the TargetedNetwork object, PhysicalNetwork objects, and VirtualNetwork objects.

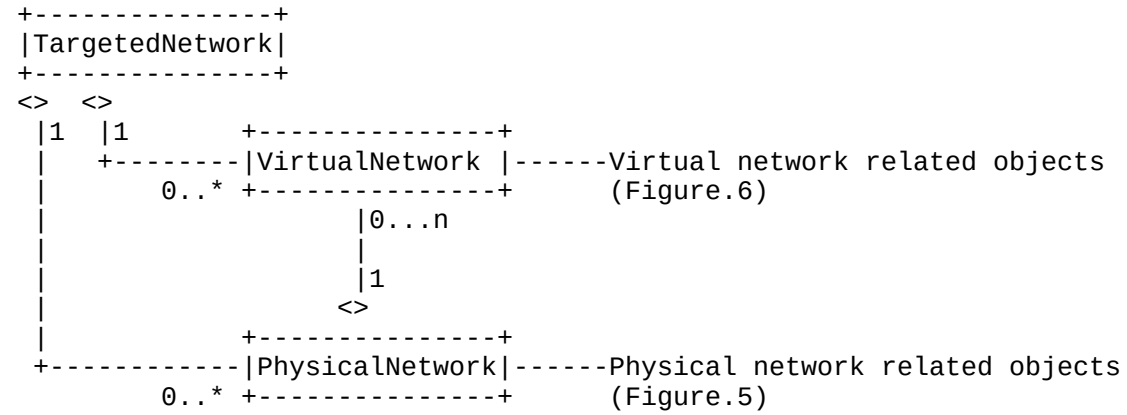

Figure 4: Class diagram of the proposed virtual-network management information model

7.2. PhysicalNetwork Object

To represent the structure of a physical network, the proposed model defines the following six types of managed objects under the TargetedNetwork object.

- PhysicalNetwork
- PhysicalNode
- PhysicalNodeGroup
- PhysicalInterface
- PhysicalInterfaceGroup
- PhysicalLink

PhysicalNetwork:
: This object represents an actual network composed of actual devices. This object aggregates zero or more PhysicalNode objects.

PhysicalNode:
: This object represents an actual device in a physical network. The actual device is a server, a server virtualization platform, or a network switch. The object has an association with a PhysicalNetwork object. It also has an association with a PhysicalNodeGroup object when the actual device is a member of a group of devices. It also aggregates zero or more PhysicalInterface objects. The PhysicalNode object can contain one "Configurations" object, which stores configuration data of the device represented by the PhysicalNode object. The Configurations object contains, for example, virtual LAN (VLAN) configuration, link aggregation (LAG) configuration or server virtualization configuration. Although this memo defines the Configurations object as a child object of the PhysicalNode object, defining the model for the configuration information is out of scope of this memo. The main reason is that the model of the Configurations object differs from one device to another.

PhysicalNodeGroup:
This object represents a set of multiple actual devices. For example, this object represents the chassis of a blade server, which includes multiple server blades and multiple network switches. This object aggregates one or more PhysicalNode objects.

PhysicalInterface:
This object represents an actual network interface of an actual device. The network interface is a port of a network interface card equipped in a server or a port of a network switch. The object also represents an internal network interface used to connect a server blade and an internal switch in a blade server. This object has an association with a PhysicalNode object. This object also has an association with a PhysicalInterfaceGroup object when the network interface is a port of the line card represented by the PhysicalInterfaceGroup object. This object also has an association with a PhysicalLink object when the network interface is connected to another network interface by an actual network cable.

PhysicalInterfaceGroup:
This object represents a set of actual network interfaces. For example, it represents a network interface card or a network switch's line card (which is equipped with multiple ports). It aggregates one or more PhysicalInterface objects.

PhysicalLink:
This object represents an actual network cable used to connect two actual network interfaces. For example, it represents a generic Ethernet cable. It also represents an internal connection between a server blade and an internal switch in a blade server. This object aggregates two PhysicalInterface objects.

Figure 5 shows an abstract class diagram of the objects related to the physical network.

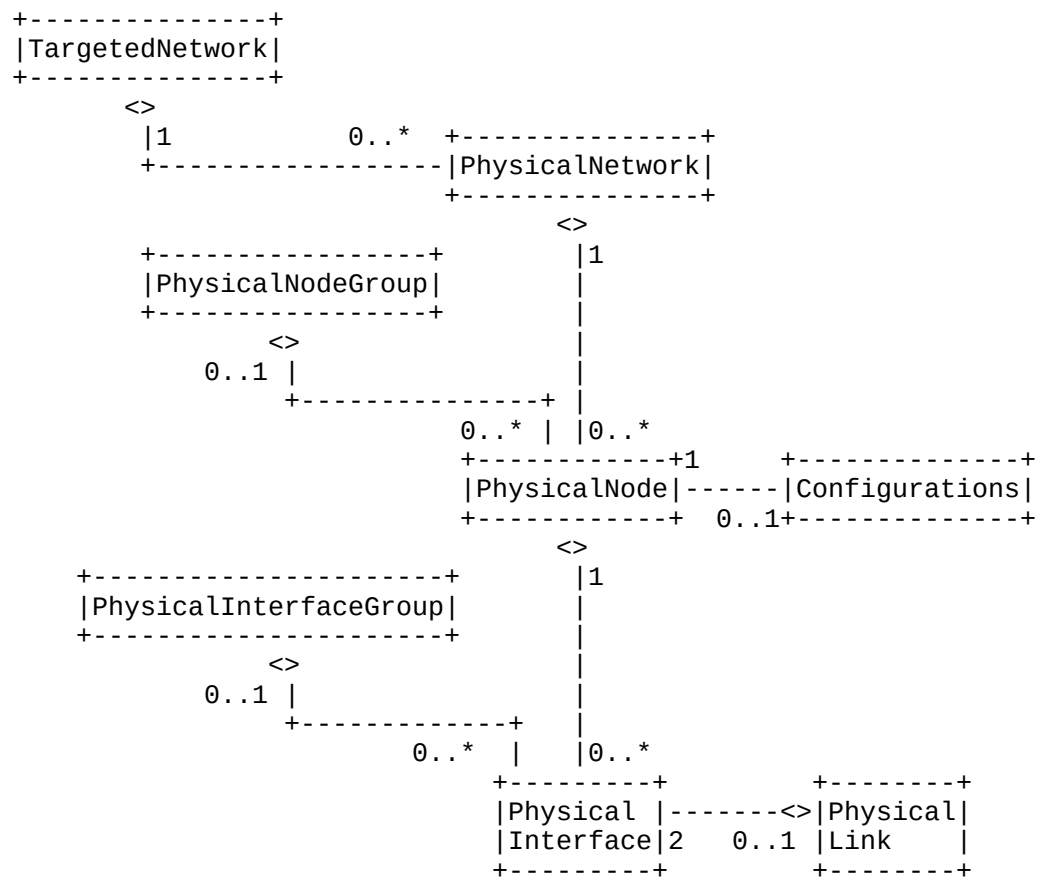

Figure 5: Class diagram of physical-network-related objects

7.3. VirtualNetwork Object

To represent the structure of a virtual network, the proposed model defines the following five types of managed objects under the TargetedNetwork object.

- VirtualNetwork
- VirtualNode
- VirtualNodeGroup
- VirtualInterface

o VirtualLink

VirtualNetwork:
This object represents a virtual network composed of multiple virtual network devices, including not only actual devices but also virtual devices. It aggregates zero or more VirtualNode objects.

VirtualNode:
This object represents a virtual network device in a virtual network. Examples of the virtual devices are virtual switches and virtual machines on a server virtualization platform. Other examples are virtual-router functions configured on a router. The object has an association with a VirtualNetwork object and a VirtualNodeGroup object.

VirtualNodeGroup:
This object represents a set of virtual devices that are created from the same actual device. It aggregates one or more VirtualNode objects. It also has an association with a PhysicalNode object, which represents an actual device.

VirtualInterface:
This object represents a virtual network interface of a virtual device. An example of such an interface is a virtual network-interface card (VNIC) of a virtual machine on a server virtualization platform. This object has an association with a VirtualNode object. This object also has an association with a VirtualLink object when the virtual network interface is connected to another virtual network interface by a virtual network link.

VirtualLink:
This object represents a virtual network link used to connect two virtual network interfaces. For example, it represents a connection between a virtual machine and a virtual switch created on a server virtualization platform. This object aggregates two VirtualInterface objects.

The relationship between the VirtualNetwork, the VirtualNode, the VirtualInterface, and this VirtualLink object is almost the same as the relationship between the PhysicalNetwork, the PhysicalNode, the PhysicalInterface, and the PhysicalLink object.

Figure 6 shows an abstract class diagram of the objects related to the virtual network.

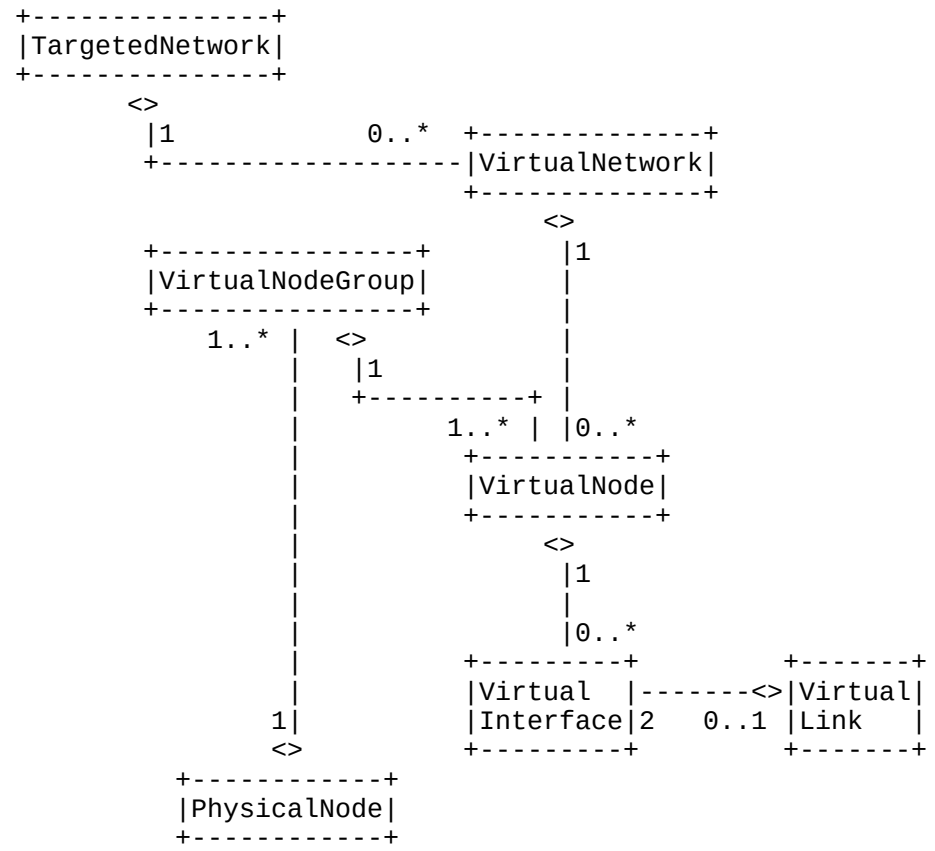

Figure 6: Class diagram of virtual-network-related objects

7.4. Id Object

All objects except the TargetedNetwork object must contain each "id" object which stores an identifier (ID). The ID must be unique within the group formed by the same type of objects associated with the same parent object as following.

- PhysicalNetwork object ID is unique within a TargetedNetwork object.
- PhysicalNodeGroup object ID is unique within a PhysicalNetwork object.
- PhysicalNode object ID is unique within a PhysicalNetwork object.

- PhysicalInterface object ID is unique within a PhysicalNode object.
- PhysicalInterfaceGroup object ID is unique within a PhysicalNode object.
- PhysicalLink object ID is unique within a PhysicalNetwork object.
- VirtualNetwork object ID is unique within a TargetedNetwork object.
- VirtualNode object ID is unique within a VirtualNetwork object.
- VirtualInterface object ID is unique within a VirtualNode object.
- VirtualLink object ID is unique within a VirtualNetwork object

8. XML-based Implementation of the Proposed Information Model

This section shows an example data model that is created according to the proposed information model described above. This example data model is intended to help readers check the feasibility of the proposed information model. Thus, this section will be removed when the proposed information model is fixed.

This example data model is defined as an XML[W3C.REC-xml-20081126]-based data model. Therefore, it is represented as an XML tree, which has an "targetedNetwork" element as its top node. In this XML tree, each class in the proposed information model is mapped to an XML element and located hierarchically.

Because of the difference between UML and XML, several new objects exist in the example XML data model. For example, a "physicalLinks" element appeared under a "physicalNetwork" element in order to aggregate multiple "physicalLink" elements. To represent the reference to one of these "physicalLink" elements, a String-type "linkId" element appears in a "physicalInterface" element.

The XML below shows the definition of the example data model written in W3C XML Schema.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="http://www.hitachi.com/vnetmodel-0.1"
  elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:vnm="http://www.hitachi.com/vnetmodel-0.1">

    <xs:element name="targetedNetwork"
      type="vnm:targetedNetworkType"></xs:element>

  <xs:complexType name="targetedNetworkType">
    <xs:sequence>
      <xs:element name="physicalNetwork"
        type="vnm:physicalNetworkType"
        maxOccurs="unbounded" minOccurs="0">
      </xs:element>
      <xs:element name="virtualNetwork"
        type="vnm:virtualNetworkType"
        maxOccurs="unbounded" minOccurs="0">
      </xs:element>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string"></xs:attribute>
  </xs:complexType>
```

```
    <xs:complexType name="physicalNetworkType">
      <xs:sequence>
        <xs:element name="physicalNodeGroup"
          type="vnm:physicalNodeGroupType"
          maxOccurs="unbounded" minOccurs="0">
        </xs:element>
        <xs:element name="physicalNode" type="vnm:physicalNodeType"
          maxOccurs="unbounded" minOccurs="0">
        </xs:element>
        <xs:element name="physicalLinks" type="vnm:physicalLinksType"
          maxOccurs="1" minOccurs="0">
        </xs:element>
      </xs:sequence>
      <xs:attribute name="id" type="xs:string"></xs:attribute>
    </xs:complexType>

      <xs:complexType name="physicalNodeGroupType">
        <xs:sequence>
          <xs:element name="physicalNode" type="vnm:physicalNodeType"
            maxOccurs="unbounded" minOccurs="0"></xs:element>
          <xs:element name="physicalNodeGroup"
            type="vnm:physicalNodeGroupType"
            maxOccurs="unbounded" minOccurs="0">
          </xs:element>
        </xs:sequence>
        <xs:attribute name="id" type="xs:string"></xs:attribute>
        <xs:attribute name="type" type="xs:string"></xs:attribute>
      </xs:complexType>

      <xs:complexType name="physicalNodeType">
        <xs:sequence>
          <xs:element name="physicalInterface"
            type="vnm:physicalInterfaceType"
            maxOccurs="unbounded" minOccurs="0">
          </xs:element>
          <xs:element name="physicalInterfaceGroup"
            type="vnm:physicalInterfaceGroupType"
            maxOccurs="unbounded" minOccurs="0">
          </xs:element>
          <xs:element name="configurations" type="xs:anyType"
            maxOccurs="1" minOccurs="0">
          </xs:element>
        </xs:sequence>
        <xs:attribute name="id" type="xs:string"></xs:attribute>
        <xs:attribute name="type" type="xs:string"></xs:attribute>
      </xs:complexType>

      <xs:complexType name="physicalLinksType">
```

```
    <xs:sequence>
      <xs:element name="physicalLink" type="vnm:physicalLinkType"
        maxOccurs="unbounded" minOccurs="0"></xs:element>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="physicalInterfaceType">
    <xs:sequence>
      <xs:element name="linkId" type="xs:string"
        maxOccurs="1" minOccurs="0"></xs:element>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string"></xs:attribute>
    <xs:attribute name="type" type="xs:string"></xs:attribute>
  </xs:complexType>

  <xs:complexType name="physicalInterfaceGroupType">
    <xs:sequence>
      <xs:element name="physicalInterfaceId" type="xs:string"
        maxOccurs="unbounded" minOccurs="1">
      </xs:element>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string"></xs:attribute>
    <xs:attribute name="type" type="xs:string"></xs:attribute>
  </xs:complexType>

  <xs:complexType name="physicalLinkType">
    <xs:sequence>
      <xs:element name="physicalInterface" type="xs:string"
        maxOccurs="2" minOccurs="2"></xs:element>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string"></xs:attribute>
    <xs:attribute name="type" type="xs:string"></xs:attribute>
  </xs:complexType>

  <xs:complexType name="virtualNetworkType">
    <xs:sequence>
      <xs:element name="virtualNode" type="vnm:virtualNodeType"
        maxOccurs="unbounded" minOccurs="0">
          </xs:element>
      <xs:element name="virtualNodeGroup"
        type="vnm:virtualNodeGroupType"
        maxOccurs="unbounded" minOccurs="0">
      </xs:element>
      <xs:element name="virtualLinks" type="vnm:virtualLinksType"
        maxOccurs="1" minOccurs="0"></xs:element>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string"></xs:attribute>
  </xs:complexType>
```

```
      <xs:complexType name="virtualNodeGroupType">
        <xs:sequence>
          <xs:element name="virtualNodeId" type="xs:string"
            maxOccurs="unbounded" minOccurs="1">
          </xs:element>
          <xs:element name="physicalNodeId" type="xs:string"
            maxOccurs="1" minOccurs="1">
          </xs:element>
        </xs:sequence>
        <xs:attribute name="id" type="xs:string"></xs:attribute>
        <xs:attribute name="type" type="xs:string"></xs:attribute>
      </xs:complexType>

      <xs:complexType name="virtualLinksType">
        <xs:sequence>
          <xs:element name="virtualLink" type="vnm:virtualLinkType"
            maxOccurs="unbounded" minOccurs="1"></xs:element>
        </xs:sequence>
      </xs:complexType>

      <xs:complexType name="virtualNodeType">
        <xs:sequence>
          <xs:element name="virtualInterface"
            type="vnm:virtualInterfaceType"
              maxOccurs="unbounded" minOccurs="0">
          </xs:element>
        </xs:sequence>
        <xs:attribute name="id" type="xs:string"></xs:attribute>
        <xs:attribute name="type" type="xs:string"></xs:attribute>
      </xs:complexType>

      <xs:complexType name="virtualLinkType">
        <xs:attribute name="id" type="xs:string"></xs:attribute>
      </xs:complexType>

      <xs:complexType name="virtualInterfaceType">
        <xs:sequence>
          <xs:element name="linkId" type="xs:string"
            maxOccurs="1" minOccurs="0"></xs:element>
        </xs:sequence>
        <xs:attribute name="id" type="xs:string"></xs:attribute>
        <xs:attribute name="type" type="xs:string"></xs:attribute>
      </xs:complexType>

      <xs:element name="NewElement" type="xs:string"></xs:element>
    </xs:schema>
```

9. YANG Module for Virtual Network Information Model

The YANG module of the data model is written by using the YANG[RFC6020] and the complex-type extension[RFC6095] and is specified as follows:

```
module vnetmodel {
  namespace "urn:ietf:params:xml:ns:vnetmodel-config";
  prefix "vnetm";

  import ietf-yang-types {prefix "yang";}
  import ietf-complex-types {prefix "ct";}

  organization "OPSAWG";
  contact "toshiaki.suzuki.cs@hitachi.com";

  description "YANG Module for Virtual Network Information Model";
  revision 2011-10 {
    description " Version of draft-okita-ops-vnetmodel-05
      Change in -05:
      - Yang Data Model is added";
  }

  ct:complex-type OriginalObject {
    description "Parameters for original object";
    leaf id { type string; }
  }

  ct:complex-type TargetedNetwork {
    description "Parameters for targeted network";
    ct:extends OriginalObject;
    ct:abstract true;
    leaf id { type string; }
    ct:instance-list physicalNetwork {
      ct:instance-type PhysicalNetwork;
    }
    ct:instance-list virtualNetwork {
      ct:instance-type VirtualNetwork;
    }
  }

  ct:complex-type PhysicalNetwork {
    description "Parameters for physical network";
    ct:extends OriginalObject;
    ct:abstract true;
    leaf id { type string; }
```

```
      ct:instance-list physicalNodeGroup {
        ct:instance-type PhysicalNodeGroup;
      }
      ct:instance-list physicalNode {
        ct:instance-type PhysicalNode;
      }
      leaf-list physicalLinks {
        type instance-identifier {
          ct:instance-type PhysicalLinks ;
        }
      }
    }

    ct:complex-type PhysicalNodeGroup {
      description "Parameters for physical node group";
      ct:extends OriginalObject;
      ct:abstract true;
      leaf id { type string; }
      leaf type { type string; }
      ct:instance-list physicalNode {
        ct:instance-type PhysicalNode;
      }
      ct:instance-list physicalNodeGroup {
        ct:instance-type PhysicalNodeGroup;
      }
    }

    ct:complex-type PhysicalNode {
      description "Parameters for physical node";
      ct:extends OriginalObject;
      ct:abstract true;
      leaf id { type string; }
      leaf type { type string; }
      leaf-list physicalInterface {
        type instance-identifier {
          ct:instance-type PhysicalInterface;
        }
      }
      leaf-list physicalInterfaceGroup {
        type instance-identifier {
          ct:instance-type PhysicalInterfaceGroup;
        }
      }
      leaf-list configurations {
        type instance-identifier {
          ct:instance-type Configulations;
        }
      }
```

```
}

ct:complex-type Configulations {
  description "Parameters for configulations";
  ct:extends OriginalObject;
  ct:abstract true;
}

ct:complex-type PhysicalLinks {
  description "Parameters for physical links";
  ct:extends OriginalObject;
  leaf-list physicalLink {
    type instance-identifier {
      ct:instance-type PhysicalLink;
    }
  }
}

ct:complex-type PhysicalInterface {
  description "Parameters for physical interface";
  ct:extends OriginalObject;
  leaf id { type string; }
  leaf type { type string; }
  leaf-list linkId { type string; }
}

ct:complex-type PhysicalInterfaceGroup {
  description "Parameters for physical interface group";
  ct:extends OriginalObject;
  ct:abstract true;
  leaf id { type string; }
  leaf type { type string; }
  leaf-list physicalInterfaceId { type string; }
}

ct:complex-type PhysicalLink {
  description "Parameters for physical link";
  ct:extends OriginalObject;
  leaf id { type string; }
  leaf type { type string; }
  leaf-list physicalInterface { type string; }
}

ct:complex-type VirtualNetwork {
  description "Parameters for virtual network";
  ct:extends OriginalObject;
  ct:abstract true;
  leaf id { type string; }
```

```
      leaf-list virtualNode {
        type instance-identifier {
          ct:instance-type VirtualNode;
        }
      }
      ct:instance-list virtualNodeGroup {
        ct:instance-type VirtualNodeGroup;
      }
      leaf-list virtualLinks {
        type instance-identifier {
          ct:instance-type VirtualLinks ;
        }
      }
    }

    ct:complex-type VirtualNodeGroup {
      description "Parameters for virtual node group";
      ct:extends OriginalObject;
      ct:abstract true;
      leaf id { type string; }
      leaf type { type string; }
      leaf-list virtualNodeID { type string; }
      leaf-list physicalNodeID { type string; }
    }

    ct:complex-type VirtualLinks {
      description "Parameters for virtual links";
      ct:extends OriginalObject;
      leaf-list virtualInterface {
        type instance-identifier {
          ct:instance-type VirtualLink ;
        }
      }
    }

    ct:complex-type VirtualNode {
      description "Parameters for virtual node";
      ct:extends VirtualNetwork;
      ct:abstract true;
      leaf id { type string; }
      leaf type { type string; }
      leaf-list virtualInterface {
        type instance-identifier {
          ct:instance-type VirtualInterface;
        }
      }
    }
```

```
    ct:complex-type VirtualLink {
      description "Parameters for virtual link";
      ct:extends OriginalObject;
      leaf id { type string; }
    }

    ct:complex-type VirtualInterface {
      description "Parameters for virtual interface";
      ct:extends OriginalObject;
      leaf id { type string; }
      leaf type { type string; }
      leaf-list linkId { type string; }
    }

  }
```

10. Security Considerations

The virtual-network management information as defined in this document provides administrative information about a data center network. This information could be used to aid an attack on the network.

It is assumed that accesses to the data defined in this document are subject to appropriate access control in the network management system.

11. IANA Considerations

The document does not request any IANA action, since the proposed model is an abstract information model. However, a concrete data model based on this information model should request IANA actions if necessary.

12. References

12.1. Normative References

[IEEE.802-1AB.2005]
"Local Area Networks and Metropolitan Area Networks: Station and Media Access Control Connectivity Discovery", IEEE Standard 802.1AB, May 2005.

[RFC2737] McCloghrie, K. and A. Bierman, "Entity MIB (Version 2)", RFC 2737, December 1999.

[RFC6020] Bjorklund, M., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, October 2010.

[RFC6095] Linowski, B., Ersue, M., and S. Kuryla, "Extending YANG with Language Abstractions", RFC 6095, March 2011.

[UML] OMG, "Unified Modeling Language", September 2002, <http://www.omg.org/technology/documents/formal/uml.htm>.

12.2. Informative References

[EVB-PAR] Congdon, P., "Edge Virtual Bridging Draft PAR", September 2009, <http://www.ieee802.org/1/files/public/docs2009/new-congdon-evb-PAR5c-0909-v1.pdf>.

[PE-PAR] Pelissier, J., "Port Extension Draft PAR Proposal", September 2009, <http://www.ieee802.org/1/files/public/docs2009/new-pelissier-portextension-par5c-0909.pdf>.

[RFC3410] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", RFC 3410, December 2002.

[RFC3535] Schoenwaelder, J., "Overview of the 2002 IAB Network Management Workshop", RFC 3535, May 2003.

[VEB] Ganga, I., "Virtual Ethernet Bridging in Server end stations", September 2008, <http://www.ieee802.org/1/files/public/docs2008/new-dcb-ganga-virtual-bridging-in-server-end-stations-0908.pdf>.

[W3C.REC-xml-20081126]
Sperberg-McQueen, C., Yergeau, F., Maler, E., Paoli, J., and T. Bray, "Extensible Markup Language (XML) 1.0 (Fifth

Edition)", World Wide Web Consortium Recommendation REC-xml-20081126, November 2008, <http://www.w3.org/TR/2008/REC-xml-20081126>.

Authors' Addresses

Hideki Okita
Central Research Laboratory, Hitachi, Ltd.
292 Yoshida-cho
Totsuka-ku, Yokohama, Kanagawa 244-0817
Japan

Phone: +81-45-860-2142
Email: hideki.okita.pf@hitachi.com

Masahiro Yoshizawa
Central Research Laboratory, Hitachi, Ltd.
292 Yoshida-cho
Totsuka-ku, Yokohama, Kanagawa 244-0817
Japan

Phone: +81-45-860-2142
Email: masahiro.yoshizawa.bt@hitachi.com

Toshiaki Suzuki
Central Research Laboratory, Hitachi, Ltd.
292 Yoshida-cho
Totsuka-ku, Yokohama, Kanagawa 244-0817
Japan

Phone: +81-45-860-2177
Email: toshiaki.suzuki.cs@hitachi.com

Tomoyuki Iijima
Central Research Laboratory, Hitachi, Ltd.
292 Yoshida-cho
Totsuka-ku, Yokohama, Kanagawa 244-0817
Japan

Phone: +81-45-860-2156
Email: tomoyuki.iijima.fg@hitachi.com

Operations and Management Area Working Group
Internet Draft
Intended status: Standard Track
Expires: Sept 2011

N. So
Verizon
P. Unbehagen
Alcatel-Lu
L. Dunbar
Huawei
H.Yu
TW Telecom
J. Heinz
CenturyLink
N.Figueira
Brocade
March 7, 2011

Requirement and Framework for VPN-Oriented Cloud Services

draft-so-vpn-o-cs-00.txt

Status of this Memo

Copyright Notice

Abstract

This contribution addresses the service providers' requirements to support VPN-Oriented Cloud services. It describes the characteristics of VPN-oriented Cloud Service and specifies the requirement on how to maintain and manage the data center resources for those services.

Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119 Error! Reference source not found..

Table of Contents

1. Introduction

Layer 2 and 3 VPN services offer secure and logically dedicated connectivity among multiple sites for enterprises. VPN-oriented Cloud Service is for those VPN customers who want to offload some dedicated user data center operations such as software, compute, and storage, to the shared cloud centers. Those customers often do not feel comfortable using public Internet as the cloud center access network. They also have more restrictive requirements on what and how the virtualized cloud center resources, e.g., computing power, disk spaces, and/or application licenses, can be shared.

VPN-Oriented Cloud Services allow the VPN services to be extended into cloud data centers and to control the virtual resources sharing functions. As a network and cloud service provider, a VPN-Oriented Cloud-service product may be offered globally across multiple data centers. Some of the data centers may be owned by a network provider, while others may be owned by a partner/vendor. In addition, multiple VPN-oriented Cloud-Service products can be offered from the same data center.

VPN-Oriented Cloud Services differentiate itself from other cloud services in the following aspects:

- Strictly maintaining the secure, reliable, and logical isolation characteristics of VPN;
- Making the traditional data center services (like computing, storage space, or application licenses) as additional attributes to VPNs.
- VPN having the control on how and what data center resources to be associated with the VPN.

This draft describes the characteristics of those services, their service requirements, and the corresponding requirements to data center networks. It also describes a list of the problems that this service is causing to the network provider/operator, especially for the existing VPN customers. These issues must be addressed immediately in order for service providers to facilitate the addition of Cloud-based services to the VPNs of existing customer.

2. Terminology

DC: Data Center

VM: Virtual Machines

VPN: Virtual Private Network

3. Service definitions and requirements

There are various types of VPN-Oriented Cloud Services. Here are just some examples:

VPN-oriented cloud computing service
This refers to Virtual Machines (VMs) and/or physical servers in a cloud data center being added to a VPN customer. The VPN customer can choose different properties on the computing power, such as dedicated servers, preference on which data center to host those servers, or special VMs which are shared with a group of other VPN customers, and etc.

Any cloud data center providing the VPN-oriented computing services SHOULD be able to automatically provision and/or change the required resources based on the specified properties associated with a VPN.

VPN customers SHALL be able to automatically instantiate or remove hosts to/from the VPN's associated Virtual Machines or dedicated servers through the changing of the customer's VPN properties.

VPN-oriented cloud storage service
This refers to disk space, either virtual or actual blocks of hard drives in data centers, being added to a customer's VPN. The VPN customer SHOULD be able to choose different properties on the storage space, such as: if the content has to replicated locally or has to be replicated at geographically different locations; if the storage has to be co-located with certain hosts; or which hosts have access to the content, and etc.

These properties are strictly associated with the VPN. Any data center providing the storage space for a VPN SHOULD be able to automatically provision or change the required storage space based on the property associated with the VPN.

The VPN customer SHOULD be able to automatically add disc space or remove disc space to the VPN's associated storage through the changing of the VPN properties.

Each VPN SHALL have the ability to limit the mobility of the stored data to a certain geographic region confinement (country/state).

4. Requirements of Data Center networks in support of VPN-Oriented Cloud Services

The success of VPN services in the enterprise and the government world is largely due to its ability to virtually segregate the customer traffic at layer 2 and layer 3. The lower the layer that segregation can be maintained, the safer it is for the customers from security and privacy perspectives. Today's Data Centers use VLANs to segregate servers and traffic from different customers. Since each customer usually needs multiple zones (e.g., DMZ, Web Server zone, and etc) to place different applications, each customer usually needs multiple VLANs. Even small data centers today already consume several thousands of VLANs. Therefore, pure VLAN segregation is not enough for large data centers.

Network service providers view data center resources as added attributes to VPNs. Therefore, traffic segregation per VPN is an essential requirement to the success of VPN-oriented Cloud-Services in the enterprises and government markets. Other essential requirements include:

Requirements for extending VPNs into data center networks using VPN gateways:

- The Cloud Service associated with certain VPN(s) SHALL be transmitted over a pre-defined set of connections, and each VPN utilizing the service SHALL be transmitted over a sub-set of logical connections.
- The VPN gateway should maintain a mapping among Virtual or physical Resources, physical/logical connections, with specific VPNs.
- The VPN Gateway SHOULD be able to control the connection traffic flow and assign the dedicated virtual resources accordingly.

Independent of the L2/3 technology, e.g., TRILL, PBB, SPB, OpenFlow, and etc, used for connecting external (customer) VPNs and data center virtual resources, e.g., , each VPN SHALL be given a unique Service ID, and traffic separation SHALL be maintained per Service ID.
When a L2/3 VPN is used as the network technology connecting the external (customer) VPN and the data center virtual resources, each external VPN SHALL be mapped to a unique internal VPN.

5. Data Center Resource Management Requirements for VPN-oriented Cloud Service

Today, data center server resources are managed by data center servers' administrators or management systems, and supported by hypervisors on the servers. The entire process is invisible to the underlying networks. The data center management functions today include managing servers, instantiating hosts to VMs, managing disk space, and etc.

Traffic loading and balancing and QoS assignments for data center networks are usually not considered by Data Center's server administration systems. There shall be a way that the VPN can connect with the Data Center's server administration systems that are important to the concept and spirit of the VPN:

The resources in data center MUST be partitioned per VPN's requirements instead of the traditional partitioning per customer.
The Cloud orchestration system SHALL have the ability to dedicate a specific block of disk space per services per VPN.
If a VPN requires dedicated access to blocks of disk space, the data center disk management system SHALL allocate the required disk space per VPN and be able to let VPN automatically retrieve the identification of those disk spaces.
If a VPN specifies its associated storage space to be accessible only by certain hosts, the data center disk management system SHALL have the ability to indicate the mechanism used to prevent the unwanted data retrieval for the block of disk space after it is no longer used by the VPN, before it can be re-used by other parties.

The VPN SHALL have the ability to request dedicated L2/3 network resources within the data center such as bandwidth, priorities, and so on.
The VPN SHALL have the ability to hold the requested resources without sharing with any other parties.
The VPN's QoS assignments SHOULD be able to synchronize with the Cloud virtual resources' QoS assignments.

6. Security Requirements

VPN-Oriented Cloud Service SHOULD support a variety of security measures in securing tenancy of virtual resources such as resource locking, containment, authentication, access control, encryption, integrity measure, and etc.
The VPN-Oriented Cloud Service SHOULD allow the security to be configured end-to-end on a per VPN per-user basis. For example, the Virtual Systems MUST resource-lock resources such as memory, but must also provide a cleaning function to insure confidentiality before being reallocated.
VPN-Oriented Cloud Service for private Clouds SHOULD specify an authentication mechanism based on an authentication algorithm (MD5, HMAC-SHA-1) for both header and payload. Encryption MAY also be use to provide confidentiality.
Security boundaries MAY also be create to maintain domains of TRUSTED, UNTRUSTED, and Hybrid. Within each domain access control, techniques MAY be used to secure resources and administrative domains.

7. Other Requirements

The VPN-Oriented Cloud Service SHALL support automatic end-to-end network configuration.
The VPN-Oriented Cloud Service solution MUST have sufficient OAM mechanisms in place to allow consistent end-to-end management of the solution in existing deployed networks. The solution SHOULD use existing protocols (e.g., IEEE 802.1ag, ITU-T Y.1731, BFD) wherever possible to facilitate interoperability with existing OAM deployments.

8. IANA Considerations

9. Acknowledgments

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

10. References

Authors' Addresses

Ning So
Verizon Inc.
2400 N. Glenville Ave.,
Richardson, TX75082
ning.so@verizonbusiness.com

Paul Unbehagen
Alcatel-Lucent
8742 Lucent Boulevard
Highlands Ranch, CO 80129
paul.unbehagen@alcatel-lucent.com

Linda Dunbar
Huawei Technologies
1700 Alma Drive, Suite 500
Plano, TX 75075, USA
Linda.dunbar@huawei.com

Henry Yu
TW Telecom
10475 Park Meadows Dr.
Littleton, CO 80124
Henry.yu@twtelecom.com

John M. Heinz
CenturyLink
600 New Century PKWY
KSNCAA0420-4B116
New Century, KS 66031
john.m.heinz@centurylink.com

Norival Figueira
Brocade Networks

130 Holger Way
San Jose, CA 95134
nfigueir@brocade.com

Intellectual Property Statement

Disclaimer of Validity

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.

Network Working Group
Internet-Draft
Intended status: Experimental
Expires: August 17, 2014

A. Capello
M. Cociglio
L. Castaldelli
Telecom Italia
A. Tempia Bonda

February 13, 2014

A packet based method for passive performance monitoring
draft-tempia-opsawg-p3m-04.txt

Abstract

This document describes a passive method to perform packet loss, delay and jitter measurements on live traffic. Implementation and deployment details are also explained in order to clarify how the tools and features currently available on existing routing platforms can be used to implement the method. This method has been invented and engineered in Telecom Italia and it's currently being used in Telecom Italia's 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 http://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 August 17, 2014.

Copyright Notice

carefully, as they describe your rights and restrictions with respect to this document.

Table of Contents

1. Introduction

Nowadays, most of the traffic in Service Providers' networks carries multimedia content. Video contents are highly sensitive to packet loss [RFC2680], while interactive contents are sensitive to delay [RFC2679], and jitter [RFC3393].

In front of this scenario, Service Providers need methodologies and tools to monitor and measure network performances with an adequate accuracy, in order to constantly control the quality of experience perceived by their customers. On the other hand, performance monitoring provides useful information for improving network management (e.g. isolation of network problems, troubleshooting, etc.).

A lot of work related to OAM, that includes also performance monitoring techniques, has been done by Standards Developing Organizations: [I-D.ietf-opsawg-oam-overview] provides a good overview of existing OAM mechanisms defined in IETF, ITU-T and IEEE. Considering IETF, a lot of work has been done on fault detection and connectivity verification, while a minor effort has been dedicated so far to performance monitoring. The IPPM WG has defined standard

metrics to measure network performance; however, the methods developed in the WG mainly refer to active measurement techniques. More recently, the MPLS WG has defined mechanisms for measuring packet loss, one-way and two-way delay, and delay variation in MPLS networks[RFC6374], but their applicability to passive measurements has some limitations, especially for pure connection-less networks.

The lack of adequate tools to measure packet loss with the desired accuracy drove an effort in Telecom Italia to design a new method for the performance monitoring of live traffic, possibly easy to implement and deploy. The effort led to the method described in this document: basically, it is a passive performance monitoring technique, potentially applicable to any kind of packet based traffic, including Ethernet, IP, and MPLS, both unicast and multicast. The method addresses primarily packet loss measurement, but it can be easily extended to one-way delay and delay variation measurements as well. It doesn’t require any protocol extension or interaction with existing protocols, thus avoiding any interoperability issue. Even if the method doesn’t raise any specific need for standardization, it could be further improved by means of some extension to existing protocols, but this aspect is left for further study and it is out of the scope of this document.

The method has been explicitly designed for passive measurements but it can also be used with active probes. Passive measurements are usually more easily understood by customers and provide a much better accuracy, especially for packet loss measurements.

The method described in this document has been invented and engineered in Telecom Italia and it’s currently being used in Telecom Italia’s network.

This document is organized as follows:

- Section 2 gives an overview of the method, including a comparison with alternate measurement strategies;
- Section 3 describes the method in detail
- Section 4 discusses implementation and deployment considerations, with special regard to the choices adopted in Telecom Italia’s own implementation;
- Section 5 includes some considerations about security aspects;
- Section 6 finally summarizes some concluding remarks.

2. Overview of the method

In order to perform packet loss measurements on a live traffic flow, different approaches exist. The most intuitive one consists in numbering the packets, so that each router that receives the flow can immediately detect a packet missing. This approach, though very simple in theory, is not simple to achieve: it requires the insertion of a sequence number into each packet and the devices must be able to extract the number and check it in real time. Such a task can be difficult to implement on live traffic: if UDP is used as the transport protocol, the sequence number is not available; on the other hand, if a higher layer sequence number (e.g. in the RTP header) is used, extracting that information from each packet and process it in real time could overload the device.

An alternate approach is to count the number of packets sent on one end, the number of packets received on the other end, and to compare the two values. This operation is much simpler to implement, but requires that the devices performing the measurement are in sync: in order to compare two counters it is required that they refer exactly to the same set of packets. Since a flow is continuous and cannot be stopped when a counter has to be read, it could be difficult to determine exactly when to read the counter. A possible solution to overcome this problem is to virtually split the flow in consecutive blocks by inserting periodically a delimiter so that each counter refers exactly to the same block of packets. The delimiter could be for example a special packet inserted artificially into the flow. However, delimiting the flow using specific packets has some limitations. First, it requires generating additional packets within the flow and requires the equipment to be able to process those packets. In addition, the method is vulnerable to out of order reception of delimiting packets and, to a lesser extent, to their loss.

The method proposed in this document follows the second approach, but it doesn’t use additional packets to virtually split the flow in blocks. Instead, it "colors" the packets so that the packets belonging to the same block will have the same color, whilst consecutive blocks will have different colors. Each change of color represents a sort of auto-synchronization signal that guarantees the consistency of measurements taken by different devices along the path.

Figure 1 represents a very simple network and shows how the method can be used to measure packet loss on different network segments: by enabling the measurement on several interfaces along the path, it is possible to perform link monitoring, node monitoring or end-to-end monitoring. The method is flexible enough to measure packet loss on

any segment of the network and can be used to isolate the faulty element.

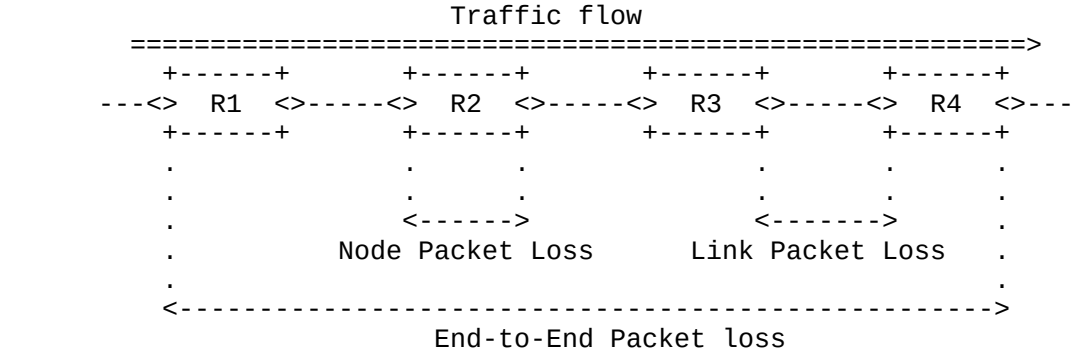

Figure 1: Available measurements

3. Detailed description of the method

This section describes in detail how the method. A special emphasis is given to the measurement of packet loss, that represents the core application of the method, but applicability to delay and jitter measurements is also considered.

3.1. Packet loss measurement

The basic idea is to virtually split traffic flows into consecutive blocks: each block represents a measurable entity unambiguously recognizable by all network devices along the path. By counting the number of packets in each block and comparing the values measured by different network devices along the path, it is possible to measure packet loss occurred in any single block between any two points.

As discussed in the previous section, a simple way to create the blocks is to "color" the traffic (two colors are sufficient) so that packets belonging to different consecutive blocks will have different colors. Whenever the color changes, the previous block terminates and the new one begins. Hence, all the packets belonging to the same block will have the same color and packets of different consecutive blocks will have different colors. The number of packets in each block depends on the criterion used to create the blocks: if the color is switched after a fixed number of packets, then each block will contain the same number of packets (except for any losses); but if the color is switched according to a fixed timer, then the number of packets may be different in each block depending on the packet rate.

The following figure shows how a flow looks like when it is split in traffic blocks with colored packets.

A: packet with A coloring
B: packet with B coloring

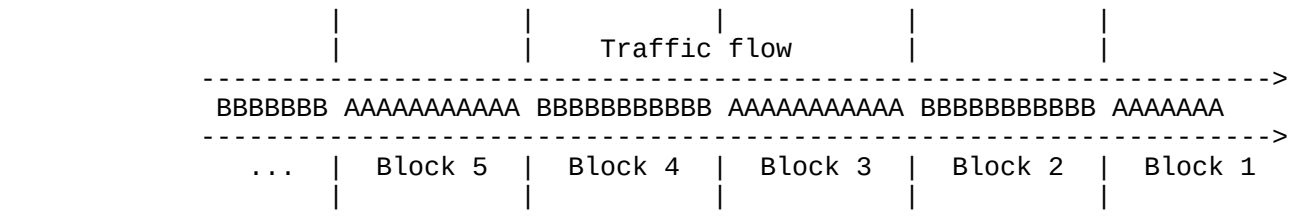

Figure 2: Traffic coloring

Figure 3 shows how the method can be used to measure link packet loss between two adjacent nodes.

Referring to the figure, let’s assume we want to monitor the packet loss on the link between two routers: router R1 and router R2. According to the method, the traffic is colored alternatively with two different colors, A and B. Whenever the color changes, the transition generates a sort of square-wave signal, as depicted in the following figure.

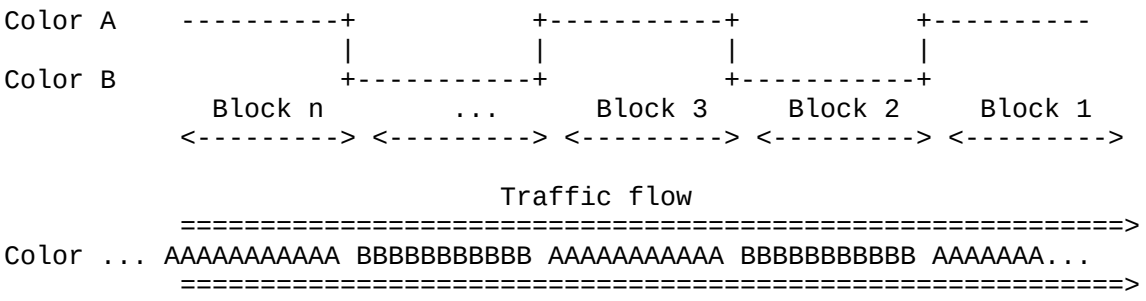

Figure 3: Application of the method to compute link packet loss

Traffic coloring could be done by R1 itself or by an upward router. R1 needs two counters, C(A)R1 and C(B)R1, on its egress interface: C(A)R1 counts the packets with color A and C(B)R1 counts those with color B. As long as traffic is colored A, only counter C(A)R1 will be incremented, while C(B)R1 is not incremented; vice versa, when the traffic is colored as B, only C(B)R1 is incremented. C(A)R1 and C(B)R1 can be used as reference values to determine the packet loss

from R1 to any other measurement point down the path. Router R2, similarly, will need two counters on its ingress interface, C(A)R2 and C(B)R2, to count the packets received on that interface and colored with color A and B respectively. When an A block ends, it is possible to compare C(A)R1 and C(A)R2 and calculate the packet loss within the block; similarly, when the successive B block terminates, it is possible to compare C(B)R1 with C(B)R2, and so on for every successive block.

Likewise, by using two counters on R2 egress interface it is possible to count the packets sent out of R2 interface and use them as reference values to calculate the packet loss from R2 to any measurement point down R2.

Using a fixed timer for color switching offers a better control over the method: the (time) length of the blocks can be chosen large enough to simplify the collection and the comparison of measures taken by different network devices. It’s preferable to read the value of the counters not immediately after the color switch: some packets could arrive out of order and increment the counter associated to the previous block (color), so it is worth waiting for some seconds. The drawback is that the longer the duration of the block, the less frequent the measurement can be taken.

The following table shows how the counters can be used to calculate the packet loss between R1 and R2. The first column lists the sequence of traffic blocks while the other columns contain the counters of A-colored packets and B-colored packets for R1 and R2. In this example, we assume that the values of the counters are reset to zero whenever a block ends and its associated counter has been read: with this assumption, the table shows only relative values, that is the exact number of packets of each color within each block. If the values of the counters were not reset, the table would contain cumulative values, but the relative values could be determined simply by difference from the value of the previous block of the same color.

The color is switched on the basis of a fixed timer (not shown in the table), so the number of packets in each block is different.

Block	C(A)R1	C(B)R1	C(A)R2	C(B)R2	Loss
1	375	0	375	0	0
2	0	388	0	388	0
3	382	0	381	0	1
4	0	377	0	374	3
...	...	...	...	...	...
n	0	387	0	387	0
n+1	379	0	377	0	2

Table 1: Evaluation of counters for packet loss measurements

During an A block (blocks 1, 3 and n+1), all the packets are A-colored, therefore the C(A) counters are incremented to the number seen on the interface, while C(B) counters are zero. Vice versa, during a B block (blocks 2, 4 and n), all the packets are B-colored: C(A) counters are zero, while C(B) counters are incremented.

When a block ends (because of color switching) the relative counters stop incrementing and it is possible to read them, compare the values measured on router R1 and R2 and calculate the packet loss within that block.

For example, looking at the table above, during the first block (A-colored), C(A)R1 and C(A)R2 have the same value (375), which corresponds to the exact number of packets of the first block (no loss). Also during the second block (B-colored) R1 and R2 counters have the same value (388), which corresponds to the number of packets of the second block (no loss). During blocks three and four, R1 and R2 counters are different, meaning that some packets have been lost: in the example, one single packet (382-381) was lost during block three and three packets (377-374) were lost during block four.

The method applied to R1 and R2 can be extended to any other router and applied to more complex networks, as far as the measurement is enabled on the path followed by the traffic flow(s) being observed.

3.2. One-way delay measurement

The same principle used to measure packet loss can be applied also to one-way delay measurement: the alternation of colors can be used as a time reference to calculate the delay. Whenever the color changes (that means that a new block has started) a network device can store the timestamp of the first packet of the new block; that timestamp can be compared with the timestamp of the same packet on a second router to compute packet delay. Considering Figure 4, R1 stores a timestamp TS(A1)R1 when it sends the first packet of block 1 (A-colored), a timestamp TS(B2)R1 when it sends the first packet of block 2 (B-colored) and so on for every other block. R2 performs the same operation on the receiving side, recording TS(A1)R2, TS(B2)R2 and so on. Since the timestamps refer to specific packets (the first packet of each block) we are sure that timestamps compared to compute delay refer to the same packets. By comparing TS(A1)R1 with TS(A1)R2 (and similarly TS(B2)R1 with TS(B2)R2 and so on) it is possible to measure the delay between R1 and R2. In order to have more measurements, it is possible to take and store more timestamps, referring to other packets within each block.

In order to coherently compare timestamps collected on different routers, the network nodes must be in sync. Furthermore, a measurement is valid only if no packet loss occurs and if packet misordering can be avoided, otherwise the first packet of a block on R1 could be different from the first packet of the same block on R2 (f.i. if that packet is lost between R1 and R2 or it arrives after the next one).

The following table shows how timestamps can be used to calculate the delay between R1 and R2. The first column lists the sequence of blocks while other columns contain the timestamp referring to the first packet of each block on R1 and R2. The delay is computed as a difference between timestamps. For the sake of simplicity, all the values are expressed in milliseconds.

Block	TS(A)R1	TS(B)R1	TS(A)R2	TS(B)R2	Delay R1-R2
1	12.483	-	15.591	-	3.108
2	-	6.263	-	9.288	3.025
3	27.556	-	30.512	-	2.956
	-	18.113	-	21.269	3.156
...	...	...	...	...	...
n	77.463	-	80.501	-	3.038
n+1	-	24.333	-	27.433	3.100

Table 2: Evaluation of timestamps for delay measurements

The first row shows timestamps taken on R1 and R2 respectively and referring to the first packet of block 1 (which is A-colored). Delay can be computed as a difference between the timestamp on R2 and the timestamp on R1. Similarly, the second row shows timestamps (in milliseconds) taken on R1 and R2 and referring to the first packet of block 2 (which is B-colored). Comparing timestamps taken on different nodes in the network and referring to the same packets (identified using the alternation of colors) it is possible to measure delay on different network segments.

For the sake of simplicity, in the above example a single measurement is provided within a block, taking into account only the first packet of each block. The number of measurements can be easily increased by considering multiple packets in the block: for instance, a timestamp could be taken every N packets, thus generating multiple delay measurements. Taking this to the limit, in principle the delay could be measured for each packet, by taking and comparing the corresponding timestamps (possible but impractical from an implementation point of view).

3.2.1. Average delay

As mentioned before, the method previously exposed for measuring the delay is sensitive to out of order reception of packets. In order to overcome this problem, a different approach has been considered: it is based on the concept of average delay. The average delay is calculated by considering the average arrival time of the packets within a single block. The network device locally stores a timestamp

for each packet received within a single block: summing all the timestamps and dividing by the total number of packets received, the average arrival time for that block of packets can be calculated. By subtracting the average arrival times of two adjacent devices it is possible to calculate the average delay between those nodes. This method is robust to out of order packets and also to packet loss (only a small error is introduced). Moreover, it greatly reduces the number of timestamps (only one per block for each network device) that have to be collected by the management system. On the other hand, it only gives one measure for the duration of the block (f.i. 5 minutes), and it doesn't give the minimum and maximum delay values. This limitation could be overcome by reducing the duration of the block (f.i. from 5 minutes to a few seconds) by means of an highly optimized implementation of the method.

By summing the average delays of the two directions of a path, it is also possible to measure the two-way delay (round-trip delay).

3.3. Delay variation measurement

Similarly to one-way delay measurement, the method can also be used to measure the inter-arrival jitter. The alternation of colors can be used as a time reference to measure delay variations. Considering the example depicted in Figure 4, R1 stores a timestamp TS(A)R1 whenever it sends the first packet of a block and R2 stores a timestamp TS(B)R2 whenever it receives the first packet of a block. The inter-arrival jitter can be easily derived from one-way delay measurement, by evaluating the delay variation of consecutive samples.

The concept of average delay can also be applied to delay variation, by evaluating the variation of consecutive measures of the average delay.

4. Implementation and deployment

The methodology described in the previous sections has been implemented in Telecom Italia by leveraging functions and tools available on IP routers and it's currently being used to monitor packet loss in some portions of Telecom Italia's network. The application of the method to delay measurement is currently being evaluated in Telecom Italia's labs.

The fundamental steps for the implementation of the method can be summarized in the following items:

- coloring the packets;

- counting the packets;

- collecting data and calculating the packet loss.

Before going deeper into the implementation details, it's worth mentioning two different strategies that can be used when implementing the method:

- flow-based: the flow-based strategy is used when only a limited number of traffic flows need to be monitored. This could be the case, for example, of IPTV channels or other specific applications traffic with high QoS requirements. According to this strategy, only a subset of the flows is colored. Counters for packet loss measurements can be instantiated for each single flow, or for the set as a whole, depending on the desired granularity. A relevant problem with this approach is the necessity to know in advance the path followed by flows that are subject to measurement. Path rerouting and traffic load-balancing increase the issue complexity, especially for unicast traffic. The problem is easier to solve for multicast traffic where load balancing is seldom used, especially for IPTV traffic where static joins are frequently used to force traffic forwarding and replication.

- link-based: measurements are performed on all the traffic on a link by link basis. The link could be a physical link or a logical link (for instance an Ethernet VLAN or a MPLS PW). Counters could be instantiated for the traffic as a whole or for each traffic class (in case it is desired to monitor each class separately), but in the second case a couple of counters is needed for each class.

The current implementation in Telecom Italia uses the first strategy. As mentioned, the flow-based measurement requires the identification of the flow to be monitored and the discovery of the path followed by the selected flow. It is possible to monitor a single flow or multiple flows grouped together, but in this case measurement is consistent only if all the flows in the group follow the same path. Moreover, a Service Provider should be aware that, if a measurement is performed by grouping many flows, it is not possible to determine exactly which flow was affected by packets loss. In order to have measures per single flow it is necessary to configure counters for each specific flow. Once the flow(s) to be monitored have been identified, it is necessary to configure the monitoring on the proper nodes. Configuring the monitoring means configuring the policy to intercept the traffic and configuring the counters to count the packets. To have just an end-to-end monitoring, it is sufficient to enable the monitoring on the first and the last hop routers of the path: the mechanism is completely transparent to intermediate nodes

and independent from the path followed by traffic flows. On the contrary, to monitor the flow on a hop-by-hop basis along its whole path it is necessary to enable the monitoring on every node from the source to the destination. In case the exact path followed by the flow is not known a priori (i.e. the flow has multiple paths to reach the destination) it is necessary to enable the monitoring system on every path: counters on interfaces traversed by the flow will report packet count, counters on other interfaces will be null.

4.1. Coloring the packets

The coloring operation is fundamental in order to create packet blocks. This implies choosing where to activate the coloring and how to color the packets.

In case of flow-based measurements, it is desirable, in general, to have a single coloring node because it is easier to manage and doesn’t rise any risk of conflict (consider the case where two nodes color the same flow). Thus it is necessary to color the flow as close as possible to the source. In addition, coloring a flow close to the source allows an end-to-end measure if a measurement point is enabled on the last-hop router as well. The only requirement is that the coloring must change periodically and every node along the path must be able to identify unambiguously the colored packets. For link-based measurements, all traffic needs to be colored when transmitted on the link. If the traffic had already been colored, then it has to be re-colored because the color must be consistent on the link. This means that each hop along the path must (re-)color the traffic; the color is not required to be consistent along different links.

Traffic coloring can be implemented by setting a specific bit in the packet header and changing the value of that bit periodically. With current router implementations, only QoS-related fields and features offer the required flexibility to explicitly set the value of some bits in the packet header from the Command Line Interface (CLI). In case a Service Provider only uses the three most significant bits of the DSCP field (corresponding to IP Precedence) for QoS classification and queuing, it is possible to use the two less significant bits of the DSCP field (bit 0 and bit 1) to implement the method without affecting QoS policies. One of the two bits (bit 0) could be used to identify flows subject to traffic monitoring (set to 1 if the flow is under monitoring, otherwise it is set to 0), while the second (bit 1) can be used for coloring the traffic (switching between values 0 and 1, corresponding to color A and B) and creating the blocks.

In practice, coloring the traffic using the DSCP field can be implemented by configuring on the router output interface an access list that intercepts the flow(s) to be monitored and applies to them a policy that sets the DSCP field accordingly. Since traffic coloring has to be switched between the two values over time, the policy needs to be modified periodically: an automatic script ca be used perform this task on the basis of a fixed timer. In Telecom Italia's implementation this timer is set to 5 minutes: this value showed to be a good compromise between measurement frequency and stability of the measurement (i.e. possibility to collect all the measures referring to the same block).

4.2. Counting the packets

Assuming that the coloring of the packets is performed only by the source node, the nodes between source and destination (included) have to count the colored packets that they receive and forward: this operation can be enabled on every router along the path or only on a subset, depending on which network segment is being monitored (a single link, a particular metro area, the backbone, the whole path).

Since the color switches periodically between two values, two counters (one for each value) are needed: one counter for packets with color A and one counter for packets with color B. For each flow (or group of flows) being monitored and for every interface where the monitoring is active, a couple of counters is needed. For example, in order to monitor separately 3 flows on a router with 4 interfaces involved, 24 counters are needed (2 counters for each of the 3 flows on each of the 4 interfaces). If traffic is colored using the DSCP field, as in Telecom Italia's implementation, an access-list that matches specific DSCP values can be used to count the packets of the flow(s) being monitored.

In case of link-based measurements the behavior is similar except that coloring and counting operations are performed on a link by link basis at each endpoint of the link.

Another important aspect to take into consideration is when to read the counters: in order to count the exact number of packets of a block the routers must perform this operation when that block has ended: in other words, the counter for color A must be read when the current block has color B, in order to be sure that the value of the counter is stable. This task can be accomplished in two ways. The general approach suggests to read the counters periodically, many times during a block duration, and to compare these successive readings: when the counter stops incrementing means that the current block has ended and its value can be elaborated safely. Alternatively, if the coloring operation is performed on the basis of

a fixed timer, it is possible to configure the reading of the counters according to that timer: for example, if each block is 5 minutes long, reading the counter for color A every 5 minute in the middle of the subsequent block (with color B) is a safe choice. A sufficient margin should be considered between the end of a block and the reading of the counter, in order to take into account any out-of-order packets. The choice of a 5 minutes timer for color switching was also suggested by these considerations

4.3. Collecting data and calculating packet loss

The nodes enabled to perform performance monitoring collect the value of the counters, but they are not able to directly use this information to measure packet loss, because they only have their own samples. For this reason, an external Network Management System (NMS) is required to collect and elaborate data and to perform packet loss calculation. The NMS compares the values of counters from different nodes and can calculate if some packets were lost (even a single packet) and also where packets were lost.

The value of the counters needs to be transmitted to the NMS as soon as it has been read. This can be accomplished by using SNMP or FTP and can be done in Push Mode or Polling Mode. In the first case, each router periodically sends the information to the NMS, in the latter case it is the NMS that periodically polls routers to collect information. In any case, the NMS has to collect all the relevant values from all the routers within one cycle of the timer (5 minutes).

5. Compliance with RFC6390 guidelines

RFC6390 [RFC6390]defines a framework and a process for developing Performance Metrics for protocols above and below the IP layer (such as IP-based applications that operate over reliable or datagram transport protocols).

This document doesn’t aim to propose a new Performance Metric but a new method of measurement for a few Performance Metrics that have already been standardized. Nevertheless, it’s worth applying RFC6390 guidelines to the present document, in order to provide a more complete and coherent description of the proposed method. We used a subset of the Performance Metric Definition template defined by RFC6390.

- Metric name and description: as already stated, this document doesn’t propose any new Performance Metric. On the contrary, it describes a novel method for measuring packet loss[RFC2680]. The same concept, with small differences, can also be used to measure

delay[RFC2679], and jitter[RFC3393]. The document mainly describes the applicability to packet loss measurement.

- Method of Measurement or Calculation: according to the method described in the previous sections, the number of packets lost is calculated by subtracting the value of the counter on the source node from the value of the counter on the destination node. Both counters must refer to the same color. The calculation is performed when the value of the counters is in a steady state.

- Units of Measurement: the method calculates and reports the exact number of packets sent by the source node and not received by the destination node.

- Measurement Points: the measurement can be performed between adjacent nodes, on a per-link basis, or along a multi-hop path, provided that the traffic under measurement follows that path. In case of a multi-hop path, the measurements can be performed both end-to-end and hop-by-hop.

- Measurement Timing: the method have a constraint on the frequency of measurements. In order to perform a measure, the counter must be in a steady state: this happens when the traffic is being colored with the alternate color; in the current implementation the time interval is set to 5 minutes.

- Implementation: the current implementation of the method uses two encodings of the DSCP field to color the packets; this enables the use of policy configurations on the router to color the packets and accordingly configure the counter for each color. The path followed by traffic being measured should be known in advance in order to configure the counters along the path and be able to compare the correct values.

- Use and Applications: the method can be used to measure packet loss with high precision (i.e. 10exp(-7)) on live traffic; moreover, by combining end-to-end and per-link measurements, the method is useful to pinpoint the single link that is experiencing loss events.

- Reporting Model: the value of the counters has to be sent to a centralized management system that perform the calculations; such samples must contain a reference to the time interval they refer to, so that the management system can perform the correct correlation; the samples have to be sent while the corresponding counter is in a steady state (within a time interval), otherwise the value of the sample should be stored locally.

- Dependencies: the values of the counters have to be correlated to the time interval they refer to; moreover, as far the current implementation is based on DSCP values, there are significant dependencies on the usage of the DSCP field: it must be possible to rely on unused DSCP values without affecting QoS-related configuration and behavior; moreover, the intermediate nodes must not change the value of the DSCP field not to alter the measurement.

- Organization of Results: the method of measurement produces singletons

- Parameters: currently, the main parameter of the method is the time interval used to alternate the colors and read the counters.

6. Security Considerations

This document specifies a method to perform measurements in the context of a Service Provider's network and has not been developed to conduct Internet measurements, so it does not directly affect Internet security nor applications which run on the Internet. However, implementation of this method must be mindful of security and privacy concerns.

There are two types of security concerns: potential harm caused by the measurements and potential harm to the measurements. For what concerns the first point, the measurements described in this document are passive, so there are no packets injected into the network causing potential harm to the network itself and to data traffic. Nevertheless, the method implies modifications on the fly to the IP header of data packets: this must be performed in a way that doesn't alter the quality of service experienced by packets subject to measurements and that preserve stability and performance of routers doing the measurements. The measurements themselves could be harmed by routers altering the coloring of the packets, or by an attacker injecting artificial traffic. Authentication techniques, such as digital signatures, may be used where appropriate to guard against injected traffic attacks.

The privacy concerns of network measurement are limited because the method only relies on information contained in the IP header without any release of user data.

7. Conclusions

The advantages of the method described in this document are:

- o easy implementation: it can be implemented using features already available on major routing platforms;
- o low computational effort: the additional load on processing is negligible;
- o accurate packet loss measurement: single packet loss granularity is achieved with a passive measurement;
- o potential applicability to any kind of packet/frame -based traffic: Ethernet, IP, MPLS, etc., both unicast and multicast;
- o robustness: the method can tolerate out of order packets and it’s not based on "special" packets whose loss could have a negative impact;
- o no interoperability issues: the features required to implement the method are available on all current routing platforms.

The method doesn’t raise any specific need for standardization, but it could be further improved by means of some extension to existing protocols. Specifically, the use of DiffServ bits for coloring the packets could not be a viable solution in some cases: a standard method to color the packets for this specific application could be beneficial.

8. IANA Considerations

There are no IANA actions required.

9. Acknowledgements

The authors would like to thank Domenico Laforgia, Daniele Accetta and Mario Bianchetti for their contribution to the definition and the implementation of the method.

10. References

10.1. Normative References

[RFC2679] Almes, G., Kalidindi, S., and M. Zekauskas, "A One-way Delay Metric for IPPM", RFC 2679, September 1999.

[RFC2680] Almes, G., Kalidindi, S., and M. Zekauskas, "A One-way Packet Loss Metric for IPPM", RFC 2680, September 1999.

[RFC3393] Demichelis, C. and P. Chimento, "IP Packet Delay Variation Metric for IP Performance Metrics (IPPM)", RFC 3393, November 2002.

10.2. Informative References

[I-D.ietf-opsawg-oam-overview]
Mizrahi, T., Sprecher, N., Bellagamba, E., and Y. Weingarten, "An Overview of Operations, Administration, and Maintenance (OAM) Tools", draft-ietf-opsawg-oam-overview-13 (work in progress), January 2014.

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

[RFC6390] Clark, A. and B. Claise, "Guidelines for Considering New Performance Metric Development", BCP 170, RFC 6390, October 2011.

Authors' Addresses

Alessandro Capello
Telecom Italia
Via Reiss Romoli, 274
Torino 10148
Italy

Email: alessandro.capello@telecomitalia.it

Mauro Cociglio
Telecom Italia
Via Reiss Romoli, 274
Torino 10148
Italy

Email: mauro.cociglio@telecomitalia.it

Luca Castaldelli
Telecom Italia
Via Reiss Romoli, 274
Torino 10148
Italy

Email: luca.castaldelli@telecomitalia.it

Alberto Tempia Bonda

Email: alberto.tempia@gmail.com