NMOP
Internet-Draft
Intended status: Experimental
Expires: 23 April 2025

T. Graf
W. Du
Swisscom
A. Huang Feng
INSA-Lyon
V. Riccobene
A. Roberto
Huawei

20 October 2024

Semantic Metadata Annotation for Network Anomaly Detection
draft-netana-nmop-network-anomaly-semantics-03

Abstract

This document explains why and how semantic metadata annotation helps to test, validate, and compare Outlier and Symptom detection, supports supervised and semi-supervised machine learning development, enables data exchange among network operators, vendors and academia and make anomalies for humans apprehensible. The proposed semantics uniforms the network anomaly data exchange between and among operators and vendors to improve their Service Disruption Detection Systems.

a mis en forme : Surlignage

a mis en forme : Surlignage

Discussion Venues

This note is to be removed before publishing as an RFC.

Discussion of this document takes place on the Operations and Management Area Working Group Working Group mailing list (nmop@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/nmop/.

Source for this draft and an issue tracker can be found at https://github.com/network-analytics/draft-netana-nmop-network-anomaly-semantics/.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 23 April 2025.

Table of Contents

1. Introduction

[I-D.~~netana~~ietf-nmop-network-anomaly-architecture] provides an overall introduction into how anomaly detection is being applied into the IP network domain and which operational data is needed. It approaches the problem space by automating what a Network Engineer would normally do when verifying a network connectivity service. Monitor from different network plane perspectives to understand wherever one network plane affects another negatively.

In order to fine tune Service Disruption Detection as described in [I-D.netana-nmop-network-anomaly-lifecycle], the results provided as analytical data need to be reviewed by a Network Engineer. Keeping the human out of the monitoring but still involving him in the alarm verification loop.

This document describes what information is needed to understand the output of the Service Disruption Detection for a ~~Network~~ network ~~Engineer~~engineer, but also at the same time is semantically structured that it can be used for Service Disruption Detection System testing by comparing the results systematically and set a baseline for supervised machine learning which requires labeled operational data.

2. Conventions and Definitions

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

2.1. Terminology

This document makes use of the terms defined in [I-D.~~netana~~ietf-nmop-network-anomaly-architecture] and [I-D.ietf-nmop-terminology].

The following terms are used as defined in [I-D.~~netana~~ietf-nmop-network-anomaly-architecture]:

* Outlier Detection
* Service Disruption Detection
* Service Disruption Detection System

The following terms are used as defined in [I-D.ietf-nmop-terminology]:

* System
* Detect
* Event
* State
* Relevance
* Problem
* Symptom
* Cause
* Alarm

3. Observed Symptoms

~~In this section o~~Observed network Symptoms are specified and categorized according to the following scheme:

Action: Which action ~~the~~ a network node performed for a packet in the Forwarding Plane, a path or adjacency in the Control Plane or state or statistical changes in the Management Plane. For Forwarding Plane we distinguish between missing, where the drop occurred outside the measured network node, drop and on-path delay, which was measured on the network node. For Control Plane we distinguish between reachability, which refers to a change in the routing or forwarding information base (RIB/FIB) and adjacency which refers to a change in peering or link-layer resolution. For

Commenté [MB1]: Any link with the discard effort in OPSAWG?

Commenté [MB2]: What is a measured node?

Management Plane we refer to state or statistical changes on interfaces.

Reason: For each action, one or more reasons describe why this action was used. For Drops in Forwarding Plane we distinguish between Unreachable because network layer reachability information was missing, Administered because an administrator configured a rule preventing the forwarding for this packet and Corrupt where the network node was unable to determine where to forward to due to packet, software or hardware error. For on-path delay we distinguish between Minimum, Average and Maximum Delay for a given flow. For Control Plane wherever a the reachability was updated or withdrawn or the adjacency was established or teared down. For Management Plane we distinguish between interfaces states up and down, and statistical errors, discards or unknown protocol counters.

Cause: For each reason one or more ~~cause~~ causes describe ~~the cause~~ why ~~the~~a network node has chosen that action.

Table 1 consolidates for the forwarding plane a list of common Symptoms with their Actions, Reasons and Causes.

Action	Reason	Cause
Missing	Previous	Time
Drop	Unreachable	next-hop
Drop	Unreachable	link-layer
Drop	Unreachable	Time To Life expired
Drop	Unreachable	Fragmentation needed and Don't Fragment set
Drop	Administered	Access-List
Drop	Administered	Unicast Reverse Path Forwarding
Drop	Administered	Discard Route
Drop	Administered	Policed
Drop	Administered	Shaped
Drop	Corrupt	Bad Packet
Drop	Corrupt	Bad Egress Interface
Delay	Min	-
Delay	Mean	-
Delay	Max	-

Commenté [MB3]: I think we need to leverage on the discard OPSAGW model here.

Commenté [MB4]: ?

```
+---------+--------------+------------------------+
```

Table 1: Describing Symptoms and their Actions,
Reason and Cause for Forwarding Plane

Table 2 consolidates for the control plane a list of common symptoms with their actions, reasons and causes.

Action	Reason	Cause
Reachability	Update	Imported
Reachability	Update	Received
Reachability	Withdraw	Received
Reachability	Withdraw	Peer Down
Reachability	Withdraw	Suppressed
Reachability	Withdraw	Stale
Reachability	Withdraw	Route Policy Filtered
Reachability	Withdraw	Maximum Number of Prefixes Reached
Adjacency	Established	Peer
Adjacency	Established	Link-Layer
Adjacency	Locally Teared Down	Peer
Adjacency	Remotely Teared Down	Peer
Adjacency	Locally Teared Down	Link-Layer
Adjacency	Remotely Teared Down	Link-Layer
Adjacency	Locally Teared Down	Administrative
Adjacency	Remotely Teared Down	Administrative
Adjacency	Locally Teared Down	Maximum Number of Prefixes Reached
Adjacency	Remotely Teared Down	Maximum Number of Prefixes Reached
Adjacency	Locally Teared Down	Transport Connection Failed

Adjacency	Remotely Teared Down	Transport Connection Failed

Table 2: Describing Symptoms and their Actions, Reason and Cause for Control Plane

Table 3 consolidates for the management plane a list of common Symptoms with their Actions, Reasons and Causes.

Action	Reason	Cause
Interface	Up	Link-Layer
Interface	Down	Link-Layer
Interface	Errors	-
Interface	Discards	-
Interface	Unknown Protocol	-

Table 3: Describing Symptoms and their Actions, Reason and Cause for Management Plane

4. Semantic Metadata

Metadata adds additional context to data. For instance, in networks the software version of a network node where Management Plane metrics are obtained from as described in[I-D.claise-opsawg-collected-data-manifest]. Where in Semantic Metadata the meaning or ontology of the annotated data is being described. In this section a YANG model is defined in order to provide a structure for the metadata related to anomalies happening in the network. The module is intended to describe the metadata used to "annotate" the operational data collected from the network nodes, which can include time series data and logs, as well as other forms of data that is "time-bounded". The aspects discussed so far in this document are grouped under the concept of "anomaly" which represents a collection of Symptoms. The anomaly overall has a set of parameters that describe the overall behavior of the network in a given time-window including all the observed Symptoms and Outliers.

4.1. Overview of the Model for the Symptom Semantic Metadata

Figure 1 contains the YANG tree diagram [RFC8340] of the Figure 2 which augments the [I-D.netana-nmop-network-anomaly-lifecycle] defined ietf-relevant-state.

For each Symptom, the following parameters have been assigned: Action, Reason and Cause to describe the Symptom, a concern score indicating how critical the Symptom is and with Forwarding, Control and Management to which network plane the Symptom can be attributed to.

```
module: ietf-network-anomaly-symptom-cbl

  augment /rsn:relevant-state/rsn:anomalies/rsn:symptom:
    +--rw action?             string
    +--rw reason?             string
    +--rw cause?              string
    +--rw (plane)?
       +--:(forwarding)
       |  +--rw forwarding?   empty
       +--:(control)
       |  +--rw control?      empty
       +--:(management)
          +--rw management?   empty
  augment /rsn:relevant-state-notification/rsn:anomalies/rsn:symptom:
    +-- action?             string
    +-- reason?             string
    +-- cause?              string
    +-- (plane)?
       +--:(forwarding)
       |  +-- forwarding?   empty
       +--:(control)
       |  +-- control?      empty
       +--:(management)
          +-- management?   empty
```

Figure 1: YANG tree diagram for ietf-network-anomaly-symptom-cbl

The module augments the "anomaly ~~grouping~~" of the relevant-state container and the relevant-state-notification notification of ietf-relevant-state. The relevant-state container is used for modifying the Symptom data in the Postmortem system. Where the relevant-state-notification ~~notification~~ is used for messaging from the Alarm Aggregation to the Postmortem and the Alarm and Problem Management system.

```
module: ietf-relevant-state
  +--rw relevant-state
     +--rw id              yang:uuid
     +--rw description?    string
     +--rw start-time      yang:date-and-time
     +--rw end-time?       yang:date-and-time
     +--rw anomalies* [id version]
        +--rw id                          yang:uuid
        +--rw version                     yang:counter32
        +--rw state                       identityref
        +--rw description?                string
        +--rw start-time                  yang:date-and-time
        +--rw end-time?                   yang:date-and-time
        +--rw confidence-score            score
        +--rw (pattern)?
        |  +--:(drop)
        |  |  +--rw drop?                 empty
        |  +--:(spike)
        |  |  +--rw spike?                empty
        |  +--:(mean-shift)
        |  |  +--rw mean-shift?           empty
        |  +--:(seasonality-shift)
        |  |  +--rw seasonality-shift?    empty
```

Commenté [MB5]: Isn't possible to have the cause be induced by several planes. The use of choice is restrictive IMO.

Commenté [MB6]: Of which doc?

```
          |  +--:(trend)
          |  |  +--rw trend?                empty
          |  +--:(other)
          |     +--rw other?                string
          +--rw annotator!
          |  +--rw name                 string
          |  +--rw (annotator-type)?
          |     +--:(human)
          |     |  +--rw human?         empty
          |     +--:(algorithm)
          |        +--rw algorithm?     empty
          +--rw symptom!
          |  +--rw id                                 yang:uuid
          |  +--rw concern-score                      score
          |  +--rw smcblsymptom:action?               string
          |  +--rw smcblsymptom:reason?               string
          |  +--rw smcblsymptom:cause?                string
          |  +--rw (smcblsymptom:plane)?
          |     +--:(smcblsymptom:forwarding)
          |     |  +--rw smcblsymptom:forwarding?     empty
          |     +--:(smcblsymptom:control)
          |     |  +--rw smcblsymptom:control?        empty
          |     +--:(smcblsymptom:management)
          |        +--rw smcblsymptom:management?     empty
          +--rw service!
             +--rw id
             |       yang:uuid
             +--rw smtopology:vpn-service-container
             |  +--rw smtopology:vpn-service* [vpn-id]
             |     +--rw smtopology:vpn-id       string
             |     +--rw smtopology:vpn-name?    string
             |     +--rw smtopology:site-ids*    string
             +--rw smtopology:vpn-node-termination-container
                +--rw smtopology:vpn-node-termination*
                         [hostname route-distinguisher]
                   +--rw smtopology:hostname                inet:host
                   +--rw smtopology:route-distinguisher     string
                   +--rw smtopology:peer-ip*
                   |       inet:ip-address
                   +--rw smtopology:next-hop*
                   |       inet:ip-address
                   +--rw smtopology:interface-id*           int32
```

```
    notifications:
      +---n relevant-state-notification
         +--ro id              yang:uuid
         +--ro description?    string
         +--ro start-time      yang:date-and-time
         +--ro end-time?       yang:date-and-time
         +--ro anomalies* [id version]
            +--ro id                          yang:uuid
            +--ro version                     yang:counter32
            +--ro state                       identityref
            +--ro description?                string
            +--ro start-time                  yang:date-and-time
            +--ro end-time?                   yang:date-and-time
            +--ro confidence-score            score
            +--ro (pattern)?
```

```
|  +--:(drop)
|  |  +--ro drop?                  empty
|  +--:(spike)
|  |  +--ro spike?                 empty
|  +--:(mean-shift)
|  |  +--ro mean-shift?            empty
|  +--:(seasonality-shift)
|  |  +--ro seasonality-shift?     empty
|  +--:(trend)
|  |  +--ro trend?                 empty
|  +--:(other)
|     +--ro other?                 string
+--ro annotator!
|  +--ro name                 string
|  +--ro (annotator-type)?
|     +--:(human)
|     |  +--ro human?         empty
|     +--:(algorithm)
|        +--ro algorithm?     empty
+--ro symptom!
|  +--ro id                                  yang:uuid
|  +--ro concern-score                       score
|  +--ro smcblsymptom:action?                string
|  +--ro smcblsymptom:reason?                string
|  +--ro smcblsymptom:cause?                 string
|  +--ro (smcblsymptom:plane)?
|     +--:(smcblsymptom:forwarding)
|     |  +--ro smcblsymptom:forwarding?      empty
|     +--:(smcblsymptom:control)
|     |  +--ro smcblsymptom:control?         empty
|     +--:(smcblsymptom:management)
|        +--ro smcblsymptom:management?      empty
+--ro service!
   +--ro id
   |       yang:uuid
   +--ro smtopology:vpn-service-container
   |  +--ro smtopology:vpn-service* [vpn-id]
   |     +--ro smtopology:vpn-id       string
   |     +--ro smtopology:vpn-name?    string
   |     +--ro smtopology:site-ids*    string
   +--ro smtopology:vpn-node-termination-container
      +--ro smtopology:vpn-node-termination*
               [hostname route-distinguisher]
         +--ro smtopology:hostname                inet:host
         +--ro smtopology:route-distinguisher     string
         +--ro smtopology:peer-ip*
         |       inet:ip-address
         +--ro smtopology:next-hop*
         |       inet:ip-address
         +--ro smtopology:interface-id*           int32
```

Figure 2: YANG tree diagram for ietf-relevant-state

Commenté [MB7]: I would focus on the augmented part, not reproduce the full tree

4.2. YANG Module

The YANG module has one ~~typdef~~ typedef defining the score and a grouping
defining Action, Reason and Cause and how it attributes to the

Commenté [MB8]: Actually, you don't need it as it is defined in the lifecycle module.

```
   network planes.

   <CODE BEGINS> file "ietf-network-anomaly-symptom-cbl@2024-10-18.yang"
   module ietf-network-anomaly-symptom-cbl {
       yang-version 1.1;
       namespace "urn:ietf:params:xml:ns:yang:ietf-network-anomaly-
symptom-cbl";
       prefix smcblsymptom;

       import ietf-relevant-state {
           prefix rsn;
           reference
             "RFC XXX: Relevant State and Relevant State Notification";
       }

     organization "IETF NMOP (Network Management Operations) Working
Group";
     contact
       "WG Web:   <http:/tools.ietf.org/wg/netconf/>
        WG List:  <mailto:nmop@ietf.org>

        Authors:  Thomas Graf
                  <mailto:thomas.graf@swisscom.com>
                  Wanting Du
                  <mailto:wanting.du@swisscom.com>
                  Alex Huang Feng
                  <mailto:alex.huang-feng@insa-lyon.fr>
                  Vincenzo Riccobene
                  <mailto:vincenzo.riccobene@huawei-partners.com>
                  Antonio Roberto
                  <mailto:antonio.roberto@huawei.com>";
       description
           "This module defines the semantic grouping to be used by a
                    Service Disruption Detection Systems. The defined
objects is
                    used to augment the anomaly container. Describing the
                    symptoms action, reason and ~~and oncern~~concern-score.

            Copyright (c) 2024~~43~~ IETF Trust and the persons identified as
            authors of the code.  All rights reserved.

            Redistribution and use in source and binary forms, with or
            without modification, is permitted pursuant to, and subject
            to the license terms contained in, the Revised BSD License
            set forth in Section 4.c of the IETF Trust's Legal Provisions
            Relating to IETF Documents
            (https://trustee.ietf.org/license-info).

            This version of this YANG module is part of RFC XXXX; see the
RFC
            itself for full legal notices.";

       revision 2024-10-18 {
           description
             "Initial version";
           reference
             "RFC XXX: Semantic Metadata Annotation for Network Anomaly
Detection";
```

Commenté [MB9]: Maybe «smcbl» ?

```
        }
        typedef score {
          type uint8 {
            range "0 .. 100";
          }
        }

        grouping cbl-symptom {
            leaf action {
                type string;
                description "action";
            }
            leaf reason {
                type string;
                description
                    "reason";
            }
            leaf cause {
                type string;
                description
                    "cause";
            }
            choice plane {
                description
                    "Network Plane affected by the symptom";
                case forwarding {
                    leaf forwarding {
                        type empty;
                    }
                }
                case control {
                    leaf control {
                        type empty;
                    }
                }
                case management {
                    leaf management {
                        type empty;
                    }
                }
            }
        }

        augment /rsn:relevant-state/rsn:anomalies/rsn:symptom {
            uses cbl-symptom;
        }

        augment /rsn:relevant-state-notification/rsn:anomalies/rsn:symptom
{
            uses cbl-symptom;
        }

    }
    <CODE ENDS>
```

Commenté [MB10]: Already define in rsn.

Commenté [MB11]: I would avoid the use of choice here.

Commenté [MB12]: Description stmt missing

Figure 3: ietf-symptom-semantic-metadata YANG Module

5. Security Considerations

The security considerations.

6. Implementation status

This section provides pointers to existing open source implementations of this draft. Note to the RFC-editor: Please remove this before publishing.

6.1. Antagonist

A tool called Antagonist has been implemented and refined during the IETF 119 and 120 hackathons, in order to validate the application of the YANG models defined in this draft. Antagonist provides visual support for two important use cases in the scope of this document:

* the generation of a ground truth in relation to Symptoms and Problems in timeseries data

* the visual validation of results produced by automated network anomaly detection tools.

The open source code can be found here: [Antagonist]

7. Acknowledgements

The authors would like to thank Reshad Rahman for his review and valuable comment.

8. References

8.1. Normative References

[Antagonist]
Riccobene, V., Roberto, A., Du, W., Graf, T., and H. Huang Feng, "Antagonist: Anomaly tagging on historical data", <https://github.com/vriccobene/antagonist>.

[I-D.ietf-nmop-terminology]
Davis, N., Farrel, A., Graf, T., Wu, Q., and C. Yu, "Some Key Terms for Network Fault and Problem Management", Work in Progress, Internet-Draft, draft-ietf-nmop-terminology-06, 17 October 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-nmop-terminology-06>.