

A YANG Data Model for Network Incident Management

draft-ietf-nmop-network-incident-yang-08

NMOP Meeting 2026-03

Author Team (Tong Hu, Luis M. Contreras,
Qin Wu, Nigel Davis, Chong Feng)

Updates in version 08

- New version of the document covering 8 issues opened in (<https://github.com/ietf-wg-nmop/draft-ietf-nmop-network-incident-yang>).
- Fix Yanglint issue in the YANG data model. (#49,#69)
- Align with RFC8407bis section 3.8.3.1 IANA template. (#51)
- Align with YANG Module Security Considerations template. (#53)
- Capitalized the term in the section 2 (#71)
- Fix Editorial issues raised by Dan (#70)
- **Clarify the relation with draft-ietf-opsawg-scheduling-oam-tests (#73)**
- **Clarify the role of incident client/handler and responsibility distinction (#67)**
- **Clarify the relation with network anomaly architecture (#75)**

Clarify the role of incident client/Handler and responsibility distinction

Why: Where to resolve the incident?

Proposals: Distinct the role and responsibility of Incident client and incident handler.

Incident server: An entity which is responsible for detecting and reporting one network incident, performing network incident diagnosis, resolution and prediction, etc.

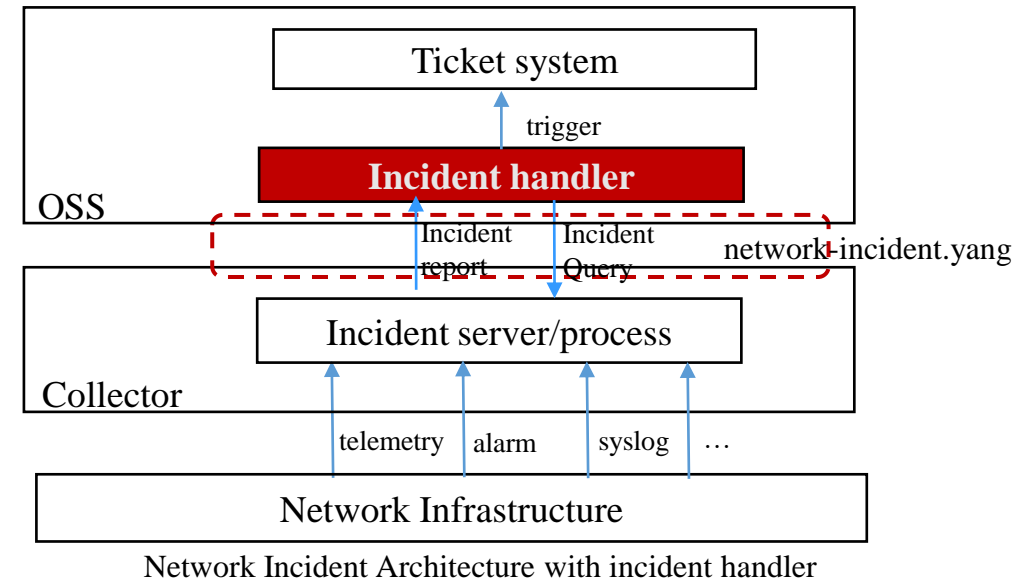
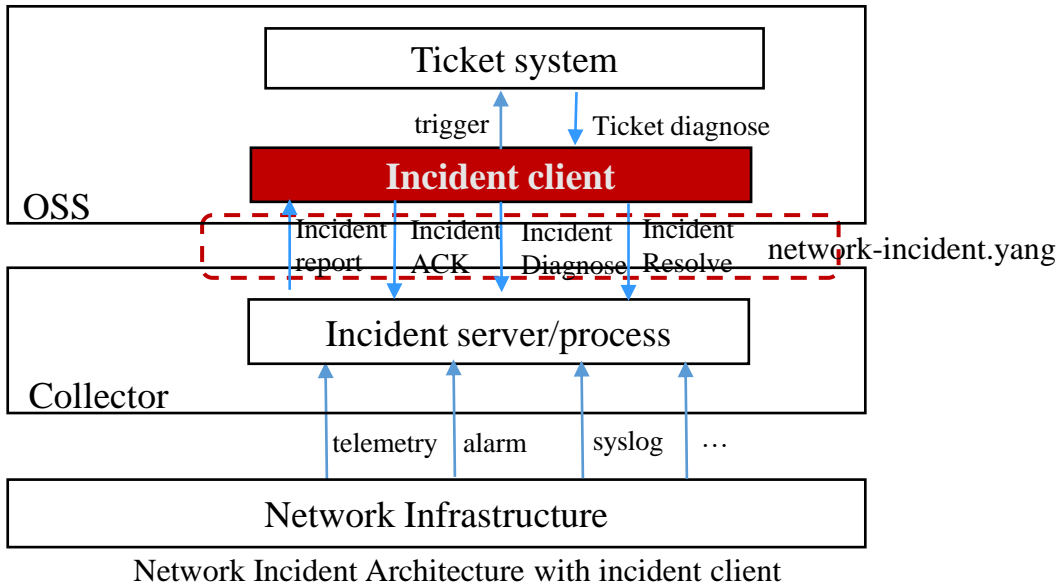
Incident client: An entity which can manage network incidents. For example, it can receive network incident notifications, query the information of network incidents, instruct an incident management server to diagnose, help resolve, etc.

Incident handler: An entity which can receive network incident notification, store and query the information of network incidents for data analysis. It has no control on incident server.

Incident server: An entity which is responsible for detecting and reporting one network incident, performing network incident diagnosis, resolution and prediction in specific domain, etc.

Incident client: An entity which can manage network incidents based on global view on network topology data correlation. For example, it can receive network incident notifications, query the information of network incidents, instruct an incident server to diagnose, help resolve, etc. In addition, it can trigger issue tickets and involve repair crew to fix the problem.

Incident handler: An entity which can receive network incident notification, store and query the information of network incidents for data analysis. Different from Incident client, it has no control on incident server or instruct an incident server to perform network incident diagnosis, resolution.



Clarify the role of incident client/Handler and responsibility distinction

Why: Where to resolve the incident?

Proposals:

1. The Incident Client has global view on network topology data
2. The incident client instructs the incident server to resolve incident

A typical workflow of network incident management is as follows:

* Some alarm report or abnormal operations, network performance metrics are reported from the network. Incident server receives

these alarms/abnormal operations/metrics and try to analyze the correlation of them, e.g., generate a symptom if some metrics are evaluated as unhealthy, the probable root cause can be detected based on the correlation analysis. If a network incident is identified, the "incident report" notification will be reported to the incident client. The impact of network services will be further analyzed and will update the network incident if the network service is impacted.

* Incident client receives the network incident from "incident report" notification raised by incident server, and acknowledge it with "incident ack" rpc operation. Client may further invoke the "incident diagnose" rpc to diagnose this network incident to find the probable root causes.

* If the probable root causes have been found, the incident client can resolve this network incident by invoking the 'incident resolve' rpc operation, dispatching a ticket or using other network functions (routing calculation, configuration, etc.)

A typical workflow of network incident lifecycle management is as follows:

* Some alarm or abnormal operations, network performance metrics, network diagnosis information [I-D.ietf-opsawg-scheduling-oam-tests] are reported from the network to the Incident Server. The Incident Server receives these alarms/abnormal operations/metrics and try to analyze the correlation of them, e.g., generate a symptom if some metrics are evaluated as unhealthy, the Probable Root Cause can be detected based on the data correlation analysis. If a network incident is identified, the "incident report" notification will be reported to the Incident Client. The impact of network services will be further analyzed and will update the network incident if the network service is impacted.

* Incident Client receives the network incident from the "incident report" notification reported by Incident Server, and acknowledges it with the subsequent "incident ack" rpc operation. The Incident Client may further invoke the "incident diagnose" rpc to diagnose this network incident to find the Probable Root Causes.

* If the Probable Root Causes have been found, the Incident Client can resolve this network incident by invoking the 'incident resolve' rpc operation to ask the Incident Server to resolve it, or dispatching a troubleshooting ticket or using other network functions (routing calculation, configuration, etc.) without being known by the Incident Server.

* In case of the 'incident resolve' rpc operation invoked by the Incident Client, the Incident Server will monitor the status of the network incident and update the status of network incident to 'cleared' if the incident can be fixed. For more detailed workflow, please refer to section 5.3.

Clarify the relation with Schedule OAM Test

Why: How Schedule OAM Test in draft-ietf-opsawg-scheduling-oam-tests can be used to support Incident Management

How: Schedule OAM Test can be used to collect network diagnosis information for data analytics

417 - * Some alarm or abnormal operations, network
performance metrics are reported from the
418 - network to the Incident Server. The Incident Server
receives these alarms/abnormal
419 - operations/metrics and try to analyze the
correlation of them, e.g., generate
420 - a symptom if some metrics are evaluated as
unhealthy, the Probable Root Cause can
421 - be detected based on the data correlation analysis.
If a network incident is identified,
422 - the "incident report" notification will be reported
to the Incident Client. The impact
423 - of network services will be further analyzed and
will update the network incident if

429 + * Some alarm or abnormal operations, network
performance metrics, network diagnosis information
430 + {{{I-D.ietf-opsawg-scheduling-oam-tests}}} are
reported from the network to the Incident Server.
431 + The Incident Server receives these alarms/abnormal
operations/metrics and try to analyze the
432 + correlation of them, e.g., generate a symptom if
some metrics are evaluated as unhealthy, the
433 + Probable Root Cause can be detected based on the
data correlation analysis. If a network incident
434 + is identified, the "incident report" notification
will be reported to the Incident Client. The
435 + impact of network services will be further analyzed
and will update the network incident if

Clarify the relation with network anomaly architecture

```

module: ietf-relevant-state
  +--rw relevant-state
    +--rw id yang:uuid
    +--rw uri? inet:uri
    +--rw description? string
    +--rw start-time yang:date-and-time
    +--rw end-time? yang:date-and-time
    +--rw strategy? string
    +--rw confidence-score? score
    +--rw concern-score score
    +--rw stage identityref
    +--rw (service)?
    +--rw anomaly* [id revision]
      +--rw id yang:uuid
      +--rw revision yang:counter32
      +--rw uri? inet:uri
      +--rw stage identityref
      +--rw description? string
      +--rw start-time yang:date-and-time
      +--rw end-time? yang:date-and-time
      +--rw confidence-score? score
      +--rw pattern? identityref
      +--rw annotator
        +--rw id? yang:uuid
        +--rw name string
        +--rw version? string
        +--rw annotator-type? enumeration
      +--rw operational-data
        +--rw topic-name? inet:host-name
        +--rw subject-name? inet:host-name
      +--rw symptom!
        +--rw id yang:uuid
        +--rw concern-score score
  
```

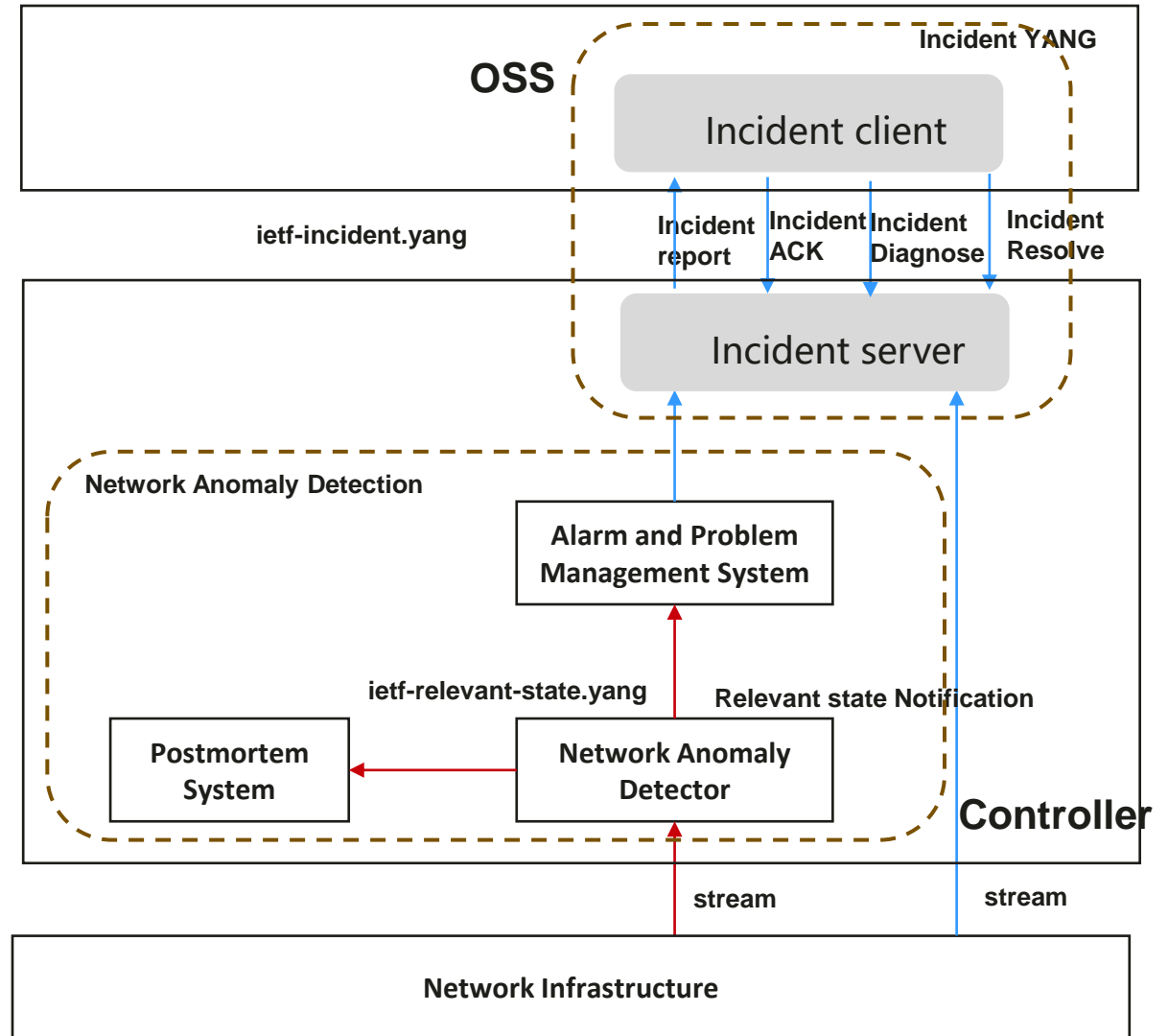
Augment by

```

module: ietf-network-anomaly-service-topology
  augment /rsn:relevant-state/rsn:service:
    +--:(l2vpn)
      +--rw l2vpn-service* [vpn-id]
        +--rw vpn-id string
        +--rw uri? inet:uri
        +--rw vpn-name? string
        +--rw site-ids* string
        +--rw change-id? yang:uuid
        +--rw change-start-time? yang:date-and-time
        +--rw change-end-time? yang:date-and-time
    +--:(l3vpn)
      +--rw l3vpn-service* [vpn-id]
        +--rw vpn-id string
        +--rw uri? inet:uri
        +--rw vpn-name? string
        +--rw site-ids* string
        +--rw change-id? yang:uuid
        +--rw change-start-time? yang:date-and-time
        +--rw change-end-time? yang:date-and-time

module: ietf-network-anomaly-symptom-cbl
  augment /rsn:relevant-state/rsn:anomaly/rsn:symptom:
    +--rw action? string
    +--rw reason? string
    +--rw trigger? string
    +--rw network-plane? enumeration
    +--rw template? string
    +--rw season? enumeration

  augment /rsn:relevant-state-notification/rsn:anomaly/rsn:symptom:
    +-- action? string
    +-- reason? string
    +-- trigger? string
    +-- network-plane? enumeration
    +-- template? string
    +-- season? enumeration
  
```

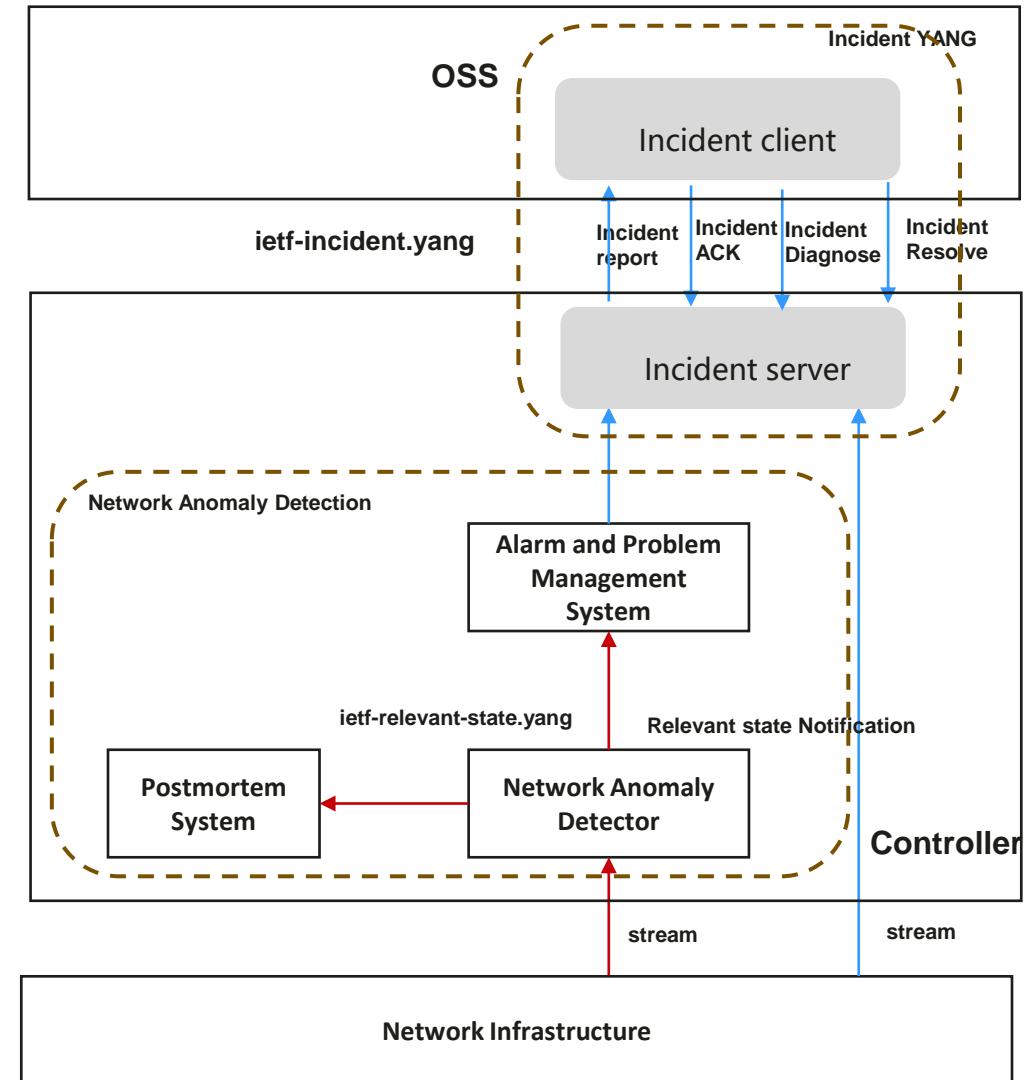


Clarify the relation with network anomaly architecture

Benoit pointed out that draft-ietf-nmop-network-anomaly-lifecycle are closely tied with this draft since:

- 3.3. Multi-layer Fault Demarcation" case to link ietf-relevant-state YANG module
- If both incident YANG module and ietf-relevant-state are deployed in the same controller, two YANG modules should refer to each other. However network anomaly detection is just one use case on how AI can be used within the controller.
- draft-ietf-nmop-network-anomaly-lifecycle Only provides one possible way to collect anomaly info And report to the incident server and incident client.
- draft-ietf-nmop-network-anomaly-lifecycle focus on the human-in-the-loop paradigm and rely on network engineers to interact with the network anomaly detection system to refine the detection abilities over time.

Proposal: make incident YANG Draft add strong dependency to draft-ietf-nmop-network-anomaly-lifecycle



Clarify the relation with network anomaly architecture

Here is the proposed solution:

Add a new subsection to clarify the relation with Network anomaly architecture

4.5. **Relationship with network anomaly architecture**

Network anomaly architecture

[\[I-D.ietf-nmop-network-anomaly-architecture\]](#) focuses on improving supervised and semi-supervised machine learning systems and evaluating anomaly detection algorithms and technologies. It also can be used to monitor network changes holistically by monitoring all 3 network planes simultaneously and detect whether that change is service disruptive. In case of disruptive changes, the anomaly can be upgraded into the network incident which trigger troubleshooting tickets generation.

Next Step

- WGLC?
- Address issue tickets raised in the GitHub
 - <https://github.com/ietf-wg-nmop/draft-ietf-nmop-network-incident-yang/issues>