A Data Manifest for Contextualized Telemetry Data
draft-claise-opsawg-collected-data-manifest-03

Abstract

Most network equipment feature telemetry as a mean to monitoring their status. Several protocols exist to this end, for example, the model-driven telemetry governed by YANG models. Some of these protocols provide the data itself, without any contextual information about the collection method. This can render the data unusable if that context is lost, for instance when the data is stored without the relevant information. This document proposes a data manifest, composed of two YANG data models, to store that contextual information along with the collected data, in order to keep the collected data exploitable in the future.

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 12 January 2023.
1. Terminology

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

Data Manifest: all the necessary data required to interpret the telemetry information.
Platform Manifest: part of the Data Manifest that completely characterizes the platform producing the data.

Data Collection Manifest: part of the Data Manifest that completely characterizes how and when the telemetry information was metered.

2. Introduction

Network elements use Model-driven Telemetry (MDT) to continuously stream information, including both counters and state information. This streamed information is used for network monitoring or closed-loop automation. This streamed data can also be stored in a database (sometimes called a big data lake) for further analysis.

When streaming YANG-structured data with YANG-Push [RFC8641], there is a semantic definition in the corresponding YANG module definition. On top of that definition, it is also important to maintain contextual information about the collection environment.

As an example, a database could store a time series representing the evolution of a specific counter. When analyzing the data, it is important to understand that this counter was requested from the network element at specific cadence, as this exact cadence might not be observed in the time series, potentially implying that the network element was under stress. The same time series might report some values as 0 or might even omit some values. This might be explained by a too small observation period, compared to the minimum-observed-period [I-D.claise-netconf-metadata-for-collection]. Again, knowing the conditions under which the counter was collected and streamed is crucial. Indeed, taking into account the value of 0 might lead to a wrong conclusion that the counter dropped to zero. This document specifies the data collection manifest, which contains the required information to characterize how and when the telemetry information was metered.

Precisely characterizing the source used for producing the data (that is the platform manifest) may also be useful to complete the data collection context. As an example, knowing the exact data source software specification might reveal a particularity in the observed data, explained by a specific bug, or a specific bug fix. This is also necessary to ensure the reliability of the collected data. On top of that, in particular for MDT, it is crucial to know the set of YANG modules supported by the device, along with their deviations. In some cases, there might even be some backwards incompatible changes in native modules between one OS version to the next one. This information must be compiled in a platform manifest.
Some related YANG modules have been specified to retrieve the device capabilities:

* [RFC9196] which models the device capabilities regarding the production and export of telemetry data.

* [I-D.claise-netconf-metadata-for-collection], which is based on the previous draft to define the optimal settings to stream specific items (i.e., per path).

While these related YANG modules are important to discover the capabilities before applying the telemetry configuration (such as on-change), some of their content is part of the context for the streamed data. The goal behind this specification is not to expose new information via YANG objects but rather to define what needs to be kept as metadata (the data manifest) to ensure that the collected data can still be interpreted correctly, even if the source device is not accessible (from the collection system), or if the device has been updated (new operating system or new configuration). This manifest contains two parts, the platform manifest and the data collection manifest. The platform manifest is "pretty" stable and should change only when the device is updated or patched. On the other hand, the data collection manifest is likely to change each time a new MDT subscription is requested and might even change if the device load increases and collection periods are updated. To separate these two parts, we enclose each of them in its own module.

We first present the module for the platform manifest in Section 3 and then the module for the data collection manifest in Section 4. The full data manifest is obtained by combining these two modules. We explain in Section 5 how the data-manifest can be collected and how collected data is mapped to the data manifest.

3. Platform Manifest

3.1. Overview of the model

Figure 1 contains the YANG tree diagram [RFC8340] of the ietf-collected-data-platform-manifest module.

module: ietf-collected-data-platform-manifest
  +--ro platform
     +--ro name?             string
     +--ro vendor?           string
     +--ro software-version? string
     +--ro software-flavor?  string
     +--ro os-version?       string
     +--ro os-type?          string
The platform manifest contains a comprehensive set of information characterize a data source. The platform is identified by a set of parameters ('name', 'software-version', 'software-flavor', 'os-
version’, ‘os-type’) that are aligned with the YANG Catalog
www.yangcatalog.org [I-D.clacla-netmod-model-catalog] so that the
YANG catalog could be used to retrieve the YANG modules a posteriori.

The platform manifest also includes the contents of the YANG Library
[RFC8525]. That module set is particularly useful to define the
paths, as they are based on module names. Similarly, this module
defines the available datastores, which can be referred to from the
data-manifest, if necessary. If supported by the device, fetching
metrics from a specific datastore could enable some specific use
cases: monitoring configuration before it is committed, comparing
between the configuration and operational datastore.

Alternatively, the set of available YANG modules on the device can be
described via packages-set which contains a list of references to
YANG Packages [I-D.ietf-netmod-yang-packages].

3.2. YANG module ietf-collected-data-platform-manifest

<CODE BEGINS> file "ietf-collected-data-platform-
manifest@2021-10-15.yang"

module ietf-collected-data-platform-manifest {
  yang-version 1.1;
  namespace
  prefix p-mf;

  import ietf-yang-library {
    prefix yanglib;
    reference
      "RFC8525: YANG Library";
  }
  import ietf-yang-packages {
    prefix pkgs;
    reference
      "RFC XXXX: YANG Packages.";
  }
  import ietf-yang-revisions {
    prefix rev;
    reference
      "XXXX: Updated YANG Module Revision Handling";
  }

  organization
    "IETF OPSAWG (Network Configuration) Working Group";
  contact
    "WG Web: <https://datatracker.ietf.org/wg/opsawg/>"
This module describes the platform information to be used as context of data collection from a given network element. The contents of this model must be streamed along with the data streamed from the network element so that the platform context of the data collection can be retrieved later.

The data content of this model should not change except on upgrade or patching of the device.

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 (RFC 2119) (RFC 8174) when, and only when, they appear in all capitals, as shown here.

Copyright (c) 2022 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 2021-10-15 {
  description
    "Initial revision";
  reference
    "RFC xxxx: Title to be completed";
}
```

// This is a copy-paste of the augments from the module
// ietf-yang-library-revisions.
// We just changed the path to which the data is added

```
augment "/p-mf:platform/p-mf:yang-library/p-mf:module-set"
  + "/p-mf:module" {
  description
    "Add a revision label to module information";
  leaf revision-label {
```

type rev:revision-label;

description
  "The revision label associated with this module revision.  
The label MUST match the rev:revision-label value in the  
specific revision of the module loaded in this module-set.";

reference
  "XXXX: Updated YANG Module Revision Handling; 
  Section 5.2.1, Advertising revision-label";

} 

augment 
  + "p-mf:submodule" { 
  description
    "Add a revision label to submodule information";
  leaf revision-label {
    type rev:revision-label;
    description
      "The revision label associated with this submodule revision.  
The label MUST match the rev:revision-label value in the  
specific revision of the submodule included by the module 
loaded in this module-set.";

    reference
      "XXXX: Updated YANG Module Revision Handling; 
      Section 5.2.1, Advertising revision-label";
  } 
}

augment 
  "/p-mf:platform/p-mf:yang-library/p-mf:module-set/
  + "p-mf:import-only-module" { 
  description
    "Add a revision label to module information";
  leaf revision-label {
    type rev:revision-label;
    description
      "The revision label associated with this module revision.  
The label MUST match the rev:revision-label value in the  
specific revision of the module included in this module-set.";

    reference
      "XXXX: Updated YANG Module Revision Handling; 
      Section 5.2.1, Advertising revision-label";
  } 
}

augment 
  "/p-mf:platform/p-mf:yang-library/p-mf:module-set/
  + "p-mf:import-only-module/p-mf:submodule" { 

"Add a revision label to submodule information";
leaf revision-label {
  type rev:revision-label;
  description
  "The revision label associated with this submodule revision. The label MUST match the rev:label value in the specific revision of the submodule included by the import-only-module loaded in this module-set.";
  reference
  "XXXX: Updated YANG Module Revision Handling;
  Section 5.2.1, Advertising revision-label";
}

augment "/p-mf:platform/p-mf:yang-library/p-mf:schema" {
  description
  "Augmentations to the ietf-yang-library module to indicate how deprecated and obsoleted nodes are handled for each datastore schema supported by the server.";
  leaf deprecated-nodes-implemented {
    type boolean;
    description
    "If set to true, this leaf indicates that all schema nodes with a status 'deprecated' are implemented equivalently as if they had status 'current'; otherwise deviations MUST be used to explicitly remove deprecated nodes from the schema. If this leaf is absent or set to false, then the behavior is unspecified.";
    reference
    "XXXX: Updated YANG Module Revision Handling;
    Section 5.2.2, Reporting how deprecated and obsolete nodes are handled";
  }
  leaf obsolete-nodes-absent {
    type boolean;
    description
    "If set to true, this leaf indicates that the server does not implement any status 'obsolete' schema nodes. If this leaf is absent or set to false, then the behavior is unspecified.";
    reference
    "XXXX: Updated YANG Module Revision Handling;
    Section 5.2.2, Reporting how deprecated and obsolete nodes are handled";
  }
}

container platform {
  config false;
}

description
   "Contains information about the platform that allows to identify
   and understand the individual data collection information. ";
leaf name {
   type string;
   description
   "Platform on which this module is implemented.";
}
leaf vendor {
   type string;
   description
   "Organization that implements that platform.";
}
leaf software-version {
   type string;
   description
   "Name of the version of software. With respect to most network
   device appliances, this will be the operating system version. But
   for other YANG module implementation, this would be a
   version of appliance software. Ultimately, this should
   correspond to a version string that will be recognizable by the
   consumers of the platform.";
}
leaf software-flavor {
   type string;
   description
   "A variation of a specific version where YANG model support
   may be different. Depending on the vendor, this could be a
   license, additional software component, or a feature set.";
}
leaf os-version {
   type string;
   description
   "Version of the operating system using this module. This is
   primarily useful if the software implementing the module is an
   application that requires a specific operating system
   version.";
}
leaf os-type {
   type string;
   description
   "Type of the operating system using this module. This is
   primarily useful if the software implementing the module is an
   application that requires a specific operating system type.";
}
container yang-library {
   description
   "The YANG library of the device specifying the modules available
4. Data Collection Manifest

4.1. Overview of the model

Figure 2 contains the YANG tree diagram [RFC8340] of the ietf-collected-data-manifest module.

Figure 2: YANG tree diagram for ietf-collected-data-manifest module

The data-collection container contains the information related to individual items collection. This subtree currently contains only information about MDT collection. It could be extended and extendable to represent other kinds of data collection.

MDT collection is organized in subscriptions. A given collector can subscribe to one or more subscriptions that usually contain a list of paths. Such a collector only needs the data manifest for subscriptions it subscribed to. The data manifest for MDT is organized by subscriptions as well so that a collector can select only its subscriptions.
We now have a chicken-and-egg issue if the collector collects the data-manifest via MDT and wants the data-manifest for the data-manifest subscription. First the collector will collect the actual paths that it needs in subscription A. Once it has the subscription id for A, it will need an additional subscription B for the data manifest of paths in A. Then, it would need another subscription C to fetch the data manifest for the subscription B and so on... A possible solution would be adding in the "mdt" container an additional list in that contains the data manifest for every path that is a data manifest. By including that list in subscription B, the collector would have the information about subscription B here.

The "datastore" leaf of the subscription container specifies from which datastore the YANG paths are streamed.

Within a given collection subscription, the granularity of the collection is defined by the path. Note that all devices do not support an arbitrary granularity up to the leaf, usually for performance reasons. Each path currently collected by the device should show up in the mdt-path-data-manifest list.

For each path, the collection context must be specified including:

* 'on-change': when set to true, an update is sent as soon as and only when a value changes. This is also known as Event-Driven Telemetry (EDT). When set to false, the values are sent regularly.

* 'suppress-redundancy' (only when 'on-change' is false): reduce bandwidth usage by sending a regular update only if the value is different from the previous update.

* 'requested-period' (only when 'on-change' is false): period between two updates requested by the client for this path.

* 'actual-period' (only when 'on-change' is false): actual period retained by the platform between two updates. That period could be larger than the requested one as the router can adjust it for performance reasons.

This information is crucial to understand the collected values. For instance, the 'on-change' and 'suppress-redundancy' options, if set, might remove a lot of messages from the database because values are sent only when there is a change.
4.2. YANG module ietf-collected-data-manifest

<CODE BEGINS> file "ietf-collected-data-manifest@2021-10-15.yang"

module ietf-collected-data-manifest {
  yang-version 1.1;
  namespace
  prefix data-manifest;

  import ietf-datastores {
    prefix ds;
    reference
      "RFC 8342: Network Management Datastore Architecture.";
  }

  import ietf-yang-types {
    prefix yang;
    reference
      "RFC 6991: Common YANG Data Types";
  }

  organization
    "IETF OPSAWG (Network Configuration) Working Group";

  contact
    "WG Web: <https://datatracker.ietf.org/wg/opsawg/>
    WG List: <mailto:opsawg@ietf.org>
    Author: Benoit Claise <mailto:benoit.claise@huawei.com>
    Author: Jean Quilbeuf <mailto:jean.quilbeuf@huawei.com>"

  description
    "This module describes the context of data collection from a
given network element. The contents of this model must be
streamed along with the data streamed from the network
element so that the context of the data collection can
be retrieved later.

This module must be completed with
ietf-collected-data-platform-manifest
to capture the whole context of a data collection session.

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 (RFC 2119)
(RFC 8174) when, and only when, they appear in all
capitals, as shown here.

Copyright (c) 2022 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 2021-10-15 {
  description
    "Initial revision";
  reference
    "RFC xxxx: Title to be completed";
}

container data-collection {
  config false;
  description
    "Defines the information for each collected object";
  list mdt-subscriptions {
    key "subscription-id";
    description
      "Contains the list of current subscriptions on the local device. Enables the collector to select its own subscriptions in the list.";
    leaf subscription-id {
      type uint64;
      description
        "Id of the subscription generated by the telemetry emitter. The collector can use this id to retrieve information about the collection status for the corresponding paths."
        The type is inspired by openconfig-telemetry. TODO check if ietf has telemetry modules that we could leafref to.
        path to subscription id in openconfig:
        openconfig-telemetry:telemetry-system/subscriptions/
            persistent-subscriptions/persistent-subscription/state/oid";
    }
    leaf datastore {
      type ds:datastore-ref;
      description
        "The datastore from which the data for this subscription has been collected.";
    }
  }
  list mdt-path-data-manifest {
    key "path";
  }
}
description
"Status of the collection for the given path";
leaf path {
  type yang:xpath1.0;
  description
  "The XPath context in which this XPath expression is evaluated is the datastore selected for the containing subscription object. If the datastore is not specified then the context is the operational datastore context.";
}
leaf requested-period {
  when "../on-change = 'false'";
  type uint64;
  description
  "Requested period, in millisecond, between two successive updates.";
}
leaf actual-period {
  when "../on-change = 'false'";
  type uint64;
  description
  "Period in millisecond between two successive updates actually applied by the platform at configuration time. This period can be larger than the requested period as the platform might adjust it for performance reasons.";
}
leaf on-change {
  type boolean;
  description
  "Whether the path is collected only when there is a change, i.e. Event-Driven Telemetry is enabled.";
}
leaf suppress-redundancy {
  type boolean;
  description
  "Whether the information is sent at every period or only when there is a change between two successive pollings.";
}

// we could augment here with other kind of collection items

"<CODE ENDS>"
5. Collecting Data Manifest and Mapping Data to Data Manifest

The data manifest MUST be streamed all with the data and stored along with the collected data. In case the collected data are moved to a different place (typically a database), the data manifest MUST follow the collected data. This can render the data unusable if that context is lost, for instance when the data is stored without the relevant information. The data manifest MUST be updated when the data manifest information changes (for example, when a router is upgraded), when a new telemetry subscription is configured, or when the telemetry subscription parameters change.

The data should be mapped to the data manifest. Since the data manifest will not change as frequently as the data itself, it makes sense to map several data to the same data manifest. Somehow, the collected data must include a metadata pointing to the corresponding data manifest.

The platform manifest is likely to remain the same until the device is updated. So, the platform manifest only needs to be collected once per streaming session and updated after a device reboot.

As this draft specifically focuses on giving context on data collected via streamed telemetry, we can assume that a streaming telemetry system is available. Collecting the data and platform manifests can be done either by reusing that streaming telemetry system (in-band) or using another system (out-of-band), for instance by adding headers or saving manifests into a YANG instance file [RFC9195].

We propose to reuse the existing telemetry system (in-band approach) in order to lower the efforts for implementing this draft. To enable a platform supporting streaming telemetry to also support data collection manifests, it is sufficient that this device supports the models from Section 3 and Section 4. Recall that each type of manifest has its own rough frequency update, i.e. at reboot for the platform manifest and at new subscription or CPU load variation for the data collection manifest. The data manifest MUST be streamed with the YANG-Push on-change feature [RFC8641] (also called event-driven telemetry).

5.1. Mapping Collected Data to the Data Manifest

With MDT, a particular datapoint is always associated to a path that is itself part of a subscription. In order to enable a posteriori retrieval of the data manifest associated to a datapoint, the collector must:
* keep the path in the metadata of the collected values
* collect as well the data-manifest for the subscription and path associated to the datapoint.

With this information, to retrieve the data manifest from the datapoint, the following happens:

* the path is retrieved from the datapoint metadata
* the data-manifest for that path is retrieved by looking up on the collected data-manifest.

In that scenario, the reliability of the collection of the data manifest is the same as the reliability of the data collection itself, since the data manifest is like any other data. For telemetry based on gRPC for instance, a disconnection to the server would be detected as the HTTP connection would fail.

6. Example

Below is an example of a data-manifest file:

<CODE BEGINS> file "ietf-collected-data-manifest@2021-10-15.yang"
The file above contains the data manifest for paths collected in the subscription with id 4242. The requested period for both path is this subscription was 100ms, however the status of the interface could only be collected every 10s.
7. Security Considerations

As we are reusing an existing telemetry system, the security considerations lies with the new content divulged in the new manifests. Appropriate access control must be associated to the corresponding leafs and containers.

8. IANA Considerations

This document includes no request to IANA.

9. Contributors

10. Open Issues

* Do we want to the hardware specifications, next to the OS information? How to fully characterize a virtual device? Do we need to include the vendor (as PEN for instance https://www.iana.org/assignments/enterprise-numbers/enterprise-numbers) ?

* Do we want to handle the absence of values, i.e. add information about missed collection or errors in the collection context? It could also explain why some values are missing. On the other hand, this might also be out scope.

* How do we handle other kinds of collection than MDT like netflow, SNMP, CLI? How do we map the collected data to the data-manifest?

* Align the terms with the YANG Push specifications. Ex: path to subscription (TBC)

* Better explain the on-change example.

* Regarding the inclusion of ietf-yang-library in our module, do we want to include as well the changes from ietf-yang-library-revisions? What if other information are present in the yang-library from the platform? Should we use a YANG mount to capture them as well (they would not be captured with our use of the main yang-library grouping).

* Henk: how does this interact with SBOM effort?

* Eliot: important to give integrity of the information a lot of thought. Threat model to be considered.

11. References
11.1. Normative References


11.2. Informative References


Appendix A. Changes between revisions

Version 3

Add when clause in YANG model

Fix validation errors on YANG modules

Augment YANG library to handle semantic versioning

Version 2

Alignment with YANGCatalog YANG module: name, vendor

Clarify the use of YANG instance file

Editorial improvements

Version 1

Adding more into data platform: yang packages, whole yanglib module to specify datastores

Setting the right type for periods: int64 -> uint64

Specify the origin datastore for mdt subscription

Set both models to config false

Applying text comments from Mohamed Boucadair

Adding an example of data-manifest file

Adding rationale for reusing telemetry system for collection of the manifests

Export manifest with on change telemetry as opposed to YANG instance file

Version 0

Initial version

Acknowledgements

Thanks to Mohamed Boucadair and Tianran Zhou for their reviews and comments.

Authors’ Addresses

Benoit Claise
Huawei
Email: benoit.claise@huawei.com

Jean Quilbeuf
Huawei
Email: jean.quilbeuf@huawei.com

Diego R. Lopez
Telefonica I+D
Don Ramon de la Cruz, 82
Madrid 28006
Spain
Email: diego.r.lopez@telefonica.com

Ignacio Dominguez
Universidad Politecnica de Madrid
Avenida Complutense 30
Madrid 28040
Spain
Email: i.dominguezm@upm.es

Thomas Graf
Swisscom
Binzring 17
CH-8045 Zurich
Switzerland
Email: thomas.graf@swisscom.com
Green Networking Metrics
draft-cx-green-metrics-00

Abstract

This document explains the need for network instrumentation that allows to assess the power consumption, energy efficiency, and carbon footprint associated with a network, its equipment, and the services that are provided over it. It also suggests a set of related metrics that, when provided visibility into, can help to optimize a network's energy efficiency and "greenness".

Status of This Memo

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

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

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

This Internet-Draft will expire on January 12, 2023.

Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.
1. Introduction

Climate change and the need to curb greenhouse emissions have been recognized by the United Nations and by most governments as one of the big challenges of our time. As a result, improving energy efficiency and reducing power consumption are becoming of increasing importance for society and for many industries. The networking industry is no exception.

Networks themselves consume significant amounts of energy. Therefore, the networking industry has an important role to play in meeting sustainability goals. Future networking advances will increasingly need to focus on becoming more energy-efficient and reducing carbon footprint, both for economic reasons and for reasons of corporate responsibility. This shift has already begun and sustainability is already becoming an important concern for network providers [telefonica2020].
There are many vectors along which networks can be made "greener". At its foundation, it involves network equipment itself. Making such equipment more energy-efficient is a big factor in helping networks become greener. However, opportunities also exist at the level of protocols themselves (e.g. reduction of transmission waste and enabling of rapid control loops), at the level of the overall network (e.g. path optimization under consideration of energy efficiency as a cost factor), and architecture level (e.g. placement of contents and functions) [I.D.draft-cwx-green-ps].

However, regardless of any particular approach that is chosen, in order to assess its impact, there is a need to have visibility into the actual energy consumption that is occurring and to ideally be able to attribute that consumption to its sources. As the adage goes, you cannot manage what you cannot measure. By extension, you cannot optimize what you have no visibility of. The ability to instrument networks in a way that allows for the assessment of energy consumption is hence an important enabler for potential energy optimizations, allowing to assess the effectiveness of measures that are being taken and enabling (for example) control loops that involve energy as an input. Before instrumenting, it needs to be clear, however, what the proper metrics are that network providers will be interested in and that applications will seek to optimize.

This document defines a set of metrics that allow to assess the "greenness" of networks and that form the basis for optimizing energy efficiency, carbon footprint, and environmental sustainability of networks and the services provided. These metrics are intended to serve the foundation for possible later IETF standardization activities, such as the definition of related YANG modules [RFC7950] or energy-related control protocol extensions.

Please note that throughout this document, we will be using the terms "green" and "energy efficient" interchangeably. In general, we will be use these terms in a broad sense, encompassing also carbon footprint and sustainability except when explicitly mentioned otherwise. Likewise, we treat "energy efficiency" as synonymous with "energy utilization efficiency", broadly speaking referring to the efficiency with which energy is being utilized.

2. Definitions and Acronyms

Carbon footprint: as used in this document, the amount of carbon emissions associated with the use or deployment of technology, usually directly correlated with the associated energy consumption.

CPU: Central Processing Unit
3.  Energy Metrics

In the following, we categorize energy metrics as follows:

- At the device/equipment level. This concerns aspects such as energy consumption of a device as a whole, of equipment components such as line cards or individual ports. It includes metrics that would, for example, be found in equipment data sheets.

- At the flow level. This concerns aspects about energy consumption by flows. Metrics at this level attribute energy consumption to a flow.

- At the path level. These metrics attest to the end-to-end energy efficiency of paths, attesting to their energy intensity (reflecting e.g. the amount of energy drawn when the path is selected) and taking into account, for example, whether a given path includes segments known to be energy-intensive.

- At the network level. These metrics aggregate energy consumption across a network to provide a holistic picture of the "network as a system".

3.1.  Energy Metrics related to Equipment

3.1.1.  Base Metrics

Arguably the most relevant energy metrics relate to equipment as a whole. After all, power is drawn from devices.

The power consumption of the device can be divided into the consumption of the core components (e.g. the backplane and CPU) as well as additional consumption incurred per port and line card. In [I.D.draft-manral-bmwg-power-usage], the device factors affecting power consumption are summarized: base chassis power, number of line cards, number of active ports, port settings, port utilization, implementation of packet classification of Ternary Content-Addressable Memory (TCAM) and the size of TCAM, firmware version.
Furthermore it is important to understand the difference between power consumption when a resource is idling versus when it is under load. This helps to understand the incremental cost of additional transmission versus the initial cost of transmission. Generally, the cost of the first bit could be considered very high, as it requires powering up a device, port, etc. The cost of transmission of additional bits (beyond the first) is many orders of magnitude lower. Likewise, the incremental cost of CPU and memory that will be needed to process additional packets becomes fairly negligible.

The first set of metrics corresponds to ratings of the device:

- Power consumption when idle (e.g. Watts)
- Power consumption when fully loaded (e.g. Watts)
- Power consumption at various loads: e.g. 50% utilization, 90% utilization

These metrics should be maintained for the device as a whole, and for the subcomponents: i.e. for the chassis by itself, for each line card, for each port. It should also take into account aspects such as the current memory configuration, as the overall energy consumption of a device is a function of the energy consumption of the components the system is comprised of.

The metrics could be provided by the data sheet associated with the device or they could be measured as part of a deployment. For maximum accuracy and comparability, they should reflect pre-defined environmental setting, e.g., operating temperature, relative humidity, barometric pressure. For example, ATIS (Alliance for Telecommunications Industry Solutions) [ATIS0600015.02] defines a reference environment under which to measure router power consumption: temperature of 25 celsius degree (within 3 celsius degree deviation), relative humidity of 30% to 75%, barometric pressure between 1020 and 812 mbar. In the AC power configuration, the router should be evaluated at 230 VAC or within 1% deviation, 50 or 60 Hz or within 1% deviation [Ahn2014].

The second set of metrics reflects the actual power being drawn during operation. It is the type of data that might be provided as management data. Again, it should be provided for the device as a whole, as well as for the subcomponents reflected in the device hierarchy: line cards, ports, etc.

- Current power consumption (e.g. Watts)
o Power drawn since system start (or module insertion, if at the
level of a line card, or port activation, if at the level of a
port), for the past minute (e.g. Watt hours)

The third set of metrics are derived from the earlier metrics. They
normalize the power consumption relative to the line speeds
respectively amount of traffic that is passed.

o Current power consumption / kilooctet

The fourth set of metrics reflects expectation values about
incremental energy usage. It could be relevant for use cases that
assess the cost of additional traffic. [Bolla2011] and [Ahn2014]
found that the power consumption of a router is in direct proportion
of the link utilization as well as the packet sizes.

o Incremental power per packet, per kilooctet, per gigaoctet.
  (Possible units might be pWh – pico Watt hours)

In addition to these metrics, it is conceivable to also have the
device reflect other context of relevance, such as the sustainability
rating of the power source. This could potentially be reflected
along a scale ranging from diesel-generator powered, via conventional
power grid, to renewable (powered by windmill, capture of excess
heat, etc). Also, the environmental status of the device could be
taken into consideration, such as whether it is deployed in a data
center and its share in contributing to the need for cooling. It is
conceivable to, for example, introduce corresponding metrics
indicating a "green rating" of device, and/or of the context in which
a device has been deployed.

3.1.2. Virtualization Considerations

Instrumentation should also take into account the possibility of
virtualization. This is important in particular as networking
functions may increasingly be virtualized and hosted (for example) in
a data center. Overlay networks may be formed. Likewise, many
applications expected to optimize energy consumption may be hosted on
controllers and applied to soft switches, VNFs (Virtual Network
Functions), or networking slices. The attribution of actual power
consumed to such virtualized entities is a non-trivial task. It
involves navigating layers of indirection to assess actual energy
usage and contribution by individual entities. While it would be
possible in such cases to simply revert to energy metrics of CPUs and
data centers as a whole, this loses the ability to account for those
metrics on the basis of networking decisions being made.
For example, virtualized networking functions could be hosted on containers or virtual machines which are hosted on a CPU in a data center instead of a regular network appliance such as a router or a switch, leading to very different power consumption characteristics. A data center CPU could be more power efficient and consume power more proportionally to actual CPU load. Virtualization could result in using fewer servers. [Energystar] reports that one watt-hour of energy savings at the server level results in roughly 1.9 watt-hours of facility-level energy savings by reducing energy waste in the power infrastructure and reducing energy needed to cool the waste heat produced by the server.

Instrumentation needs to reflect these facts and facilitate attributing power consumption in a correct manner. Alternatively, a simpler solution may be to simply forgo energy metrics for virtualized functions entirely, instead focus on instrumenting and relying on optimizing the energy footprint of the underlying hosting infrastructure. In the meantime, the attribution of energy consumption and carbon footprint to individual functions that run on top of that infrastructure may be a topic for further research.

3.2. Energy Metrics related to Flows

Energy metrics related to flows attempt to capture the contribution of a given flow to energy consumption. In its basic incarnation, those metrics reflect the energy consumption at a given device. They could be used in conjunction with IPFIX [RFC7011] and modeled as Information Elements to be treated analogous to other flow statistics [RFC7012]. The following is a corresponding set of flow energy metrics:

- **Incremental energy consumed over the duration of the flow.**
  This is the incremental energy consumption that is directly caused by the flow, representing the difference between the amount of energy consumed with the flow and the amount of energy that would have been consumed without the flow. (It should be noted that this metric may be difficult to assess in practice.)

- **Amortized energy consumed over the duration of the flow.**
  This is the portion of the flow's energy consumption for the duration of the flow, effectively computed by computing the proportion of flow traffic to overall traffic and multiplying it with the total energy consumption incurred for that time.

A second set of energy metrics related to flow might aggregate the flow’s energy consumption over the entire flow path. In that case, the flow energy consumption is added up along the systems of the traversed path. In practice, this will be more difficult to assess...
for many reasons, including impacts of load balancing, PREOF (Packet Replication, Elimination, and Ordering Functions [RFC8655]), challenges to trace actual routes taken by production traffic, and more.

3.3. Energy Metrics related to Paths

Energy metrics related to paths involve assessing the carbon footprints of paths and optimizing those paths so that overall footprint is minimized, then applying techniques such as path-aware networking [I.D.draft-chunduri-rtgw-preferred-path-routing] or segment routing [RFC8402] to steer traffic along those paths that are deemed "the greenest" among alternatives. It also includes aspects such as considering the incremental energy usage in routing decisions.

Optimizing cost has a long tradition in networking; many of the existing mechanisms can be leveraged for greener networking simply by introducing energy footprint as a cost factor. Low-hanging fruit include the inclusion of energy-related parameters as a cost parameter in control planes, whether distributed (e.g. IGP) or conceptually centralized via SDN controllers. In addition to power consumption over a path itself, other factors such as paths involving intermediate routers that are powered by renewable energy resources might be considered, as might be determined by an aggregate sustainability score. After all, paths with devices that are powered by solar, wind, or geothermal might be preferable over paths involving devices powered by conventional energy that may include fossil fuel or nuclear resources.

The following are a corresponding set of candidate metrics:

- Energy rating of a path. (This could be computed as a function of energy ratings of different hops along the path.)

- Current power consumption across a path. (This could be computed by aggregating the current power per packet (or per kilo octet etc) of each of the hops along the path.)

- Incremental power for a packet over a path. (This could be computed by aggregating the incremental power per packet of each of the hops along the path.)

3.4. Energy Metrics related to the Network-at-Large

Ultimately, the goal of energy optimization and reduction of carbon footprint is to minimize the aggregate amount of energy used across the entire network, as well as to minimize the overall carbon
footprint of the network as a whole. Accordingly, metrics that aggregate the energy usage across the network as a whole are needed. In order to account for changing traffic profiles, growth in user traffic etc, additional metrics are needed that normalize the total over the volume of services supported and volume of traffic passed. Corresponding metrics will generally be computed at the level of Operational Support Systems (or Business Support Systems) for the entire network.

Some of the metrics used include the following [telefonica2020]:

- Total energy consumption (MWh)
- Electricity from renewable sources (%)
- Network energy efficiency (MWh/PB)

4. Other considerations and discussion items

This document is intended to spark discussion about what energy metrics will be useful to reduce the carbon footprint of networks - that provide visibility into energy consumption, that help optimization of networks under green criteria, that enable the next generation of energy-aware controllers and services. Clearly, other metrics are conceivable and more considerations apply beyond those that are currently reflected in this document. The following subsections highlight items that warrant further discussion and that might be addressed in greater detail in future revisions of this document.

4.1. User perspective

Arguably, attributing energy usage to individual users and making users aware of the energy-implications of their communication behavior may provide interesting possibilities to reduce energy footprint by guiding their behavior accordingly. For example, the network could present clients with energy statistics related to their communication usage. This could be supported by metrics related to service instances, such as energy usage statistics beyond statistics regarding volume, duration, number of transactions. Such approaches would raise questions about how to actually collect such statistics accurately (versus just computing them via a formula) or what to actually include as part of those statistics (amortized vs incremental energy contribution, attribution of cost for path resilience or retransmissions due to congestion, etc). They also raise questions about how they would in practice be used. For example, energy-based charging might be explored as an alternative for volume-based charging; however, in practice the two may be
strongly correlated and rejected by customers for similar reasons that volume-based charging is frequently rejected.

4.2. Holistic perspective

The network itself is only one contributor to a network’s carbon footprint. Arguably just as important are aspects outside the network itself, such as cooling and ventilation. These aspects need to be considered. However, reflecting such aspects here would arguably result in "boiling the ocean" and are therefore not addressed here.

4.3. Sustainable equipment production

Internet energy consumption may constitute two major components [Raghavan2011]: (1) the energy of the devices that construct the Internet, including the infrastructure devices: routers, LAN devices, cellular and telecommunication infrastructure, (2) More broadly, with the rise of peer-to-peer applications and cloud services, it also considers the energy consumption of the end systems, including desktops, laptops, smart phones, cloud servers, and application servers that are not in the cloud.

For those two components, the following factors need to take into consideration for energy consumption calculation:

- Energy consumed in manufacturing of the devices and end-systems, as well as the contribution from their components and materials.

- The replacement lifespan of the devices and end-systems: desktops and laptops are typically replaced in 3-4 years, smartphones in 2 years, application servers and cloud servers in 3 years, routers and WiFi-LAN switches in 3 years, cellular towers and telecommunication switches in 10 years, fiber optics in 10 years, copper in 30 years, etc. With the incremental growth rate of the technology advancement, the replacement lifespan might be decreased over time.

- Operational maintenance: the network would not be functional without various software and implementation of protocols. The energy consumed in creating software is complicated because it is overwhelmingly human involved, which usually include the energy used for the facilities of the software companies and human energy of the programmers.

- Replacement: The energy consumed in replacement of devices and end-systems could vary. Some could be very energy intensive for
those large devices, e.g., cellular towers, or environmental unfriendly equipment, such as submarine communication cables.

- Disposal: There is substantial energy cost in disposing and recycling the old devices and equipment.

By combining the energy consumption for running each device that builds the Internet [JuniperRouterPower], and the energy consumption of the end systems, in the meantime counting the energy consumption of manufacturing, operational maintenance, replacement and lifespan, disposal of those devices and equipment, we may have an estimate of the energy consumption for the network as a whole.

5. IANA Considerations

This document does not have any IANA requests.

6. Security Considerations

When instrumenting a network for energy metrics, it is important that implementations are secured to ensure that data is accurately measured and cannot be tampered with. For example, an attacker might try to tamper energy readings to confuse controller trying to minimize power consumption, leading to increased power consumption instead. In addition, access to the data needs to be secured in similar ways as for other sensitive management data, for example using secure management protocols and subjecting energy data that is maintained in YANG datastores via NACM (NETCONF Access Control Model).

However, it should be noted that this draft specifies only metrics themselves, not how to instrument networks accordingly. For the definition of metrics themselves, security considerations do thus not really apply.

7. Acknowledgments

Acknowledgments will be added when the time comes.

8. Informative References

[ATIS0600015.02]

[Bolla2011]

[Energystar]

[I.D.draft-chunduri-rtgwg-preferred-path-routing]

[I.D.draft-cwx-green-ps]

[I.D.draft-manral-bmwg-power-usage]

[JuniperRouterPower]

[Raghavan2011]

[RFC7011]


Authors’ Addresses

Alexander Clemm
Futurewei
2220 Central Expressway
Santa Clara CA 95050
USA

Email: ludwig@clemm.org

Lijun Dong
Futurewei
2220 Central Expressway
Santa Clara CA 95050
USA

Email: lijun.dong@futurewei.com

Greg Mirsky
Ericsson

Email: gregimirsky@gmail.com
Challenges and Opportunities in Green Networking
draft-cx-green-ps-00

Abstract

Reducing technology's carbon footprint is one of the big challenges of our age. Networks are an enabler of applications that reduce this footprint, but also contribute to this footprint substantially themselves. The biggest opportunities to reduce the energy footprint may not be networking specific, for instance general power efficiency gains in hardware or hosting of equipment in more cooling-efficient buildings. Yet methods to make networking technology itself "greener" also need to be explored. This document outlines a corresponding set of opportunities, along with associated research challenges, for reducing this footprint and reducing network energy demand.

Status of This Memo

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

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

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

This Internet-Draft will expire on January 12, 2023.
1. Introduction

Climate change and the need to curb greenhouse emissions have been recognized by the United Nations and by most governments as one of the big challenges of our time. As a result, improving energy efficiency and reducing power consumption are becoming of increasing importance for society and for many industries. The networking industry is no exception.

Arguably, networks can already be considered "green" technology in that networks enable many applications that allow users and whole industries to save energy and become more sustainable in a significant way. For example, it allows (at least to an extent) to
replace travel with teleconferencing; it enables many employees to work from home and "telecommute," thus reducing the need for actual commute; IoT applications that facilitate automated monitoring and control from remote sites help make agriculture more sustainable by minimizing the application of resources such as water and fertilizer; networked smart buildings allow for greater energy optimization and sparser use of lighting and HVAC (heating, ventilation, air conditioning) than their non-networked not-so-smart counterparts.

That said, networks themselves consume significant amounts of energy. Therefore, the networking industry has an important role to play in meeting sustainability goals not just by enabling others to reduce their reliance on energy, but by also reducing its own. Future networking advances will increasingly need to focus on becoming more energy-efficient and reducing carbon footprint, both for economic reasons and for reasons of corporate responsibility. This shift has already begun and sustainability is already becoming an important concern for network providers. In some cases such as in the context of networked data centers, the ability to procure enough energy becomes a bottleneck prohibiting further growth and greater sustainability thus becomes a business necessity.

For example, in its annual report, Telefonica reports that in 2020, its network’s energy consumption per PB of data amounted to 78MWh [telefonica2020]. This rate has been dramatically decreasing (a five-fold factor over five years) although gains in efficiency are being offset by simultaneous growth in data volume. In the same report, it is stated as an important corporate goal to continue on that trajectory and reduce overall carbon emissions by 70% over the next 5 years.

From a technical perspective, multiple vectors along which networks can be made "greener" should be considered:

- At the equipment level. Perhaps the most promising vector for improving networking sustainability concerns the network equipment itself. At the most fundamental level, networks (even softwarized ones) involve appliances, i.e. equipment that relies on electrical power to perform its function. However, beyond making those appliances merely energy-efficient, there are other important ways in which equipment can help networks become greener. This includes aspects such as support for port power saving modes allowing to reduce power consumption for resources that are not fully utilized, but also instrumentation that allows to precisely monitor power usage at different levels of granularity, enabling (for example) controllers applications that aim to optimize energy usage across the network. (As a side note, the term "device", as used in the context of this draft, is used to refer to networking...
o At the protocol level. Energy-efficiency and greenness are aspects that are rarely considered when designing network protocols. This suggests that there may be plenty of untapped potential. Some aspects involve designing protocols in ways that reduce the need for redundant or wasteful transmission of data to allow not only for better network utilization, but greater goodput per unit of energy being consumed. Techniques include approaches that reduce the "header tax" incurred by payloads as well as methods resulting in the reduction of wasteful retransmissions. Likewise, aspects such as restructuring addresses in ways that allow to minimize the size of lookup tables and associated memory sizes and their energy use can play a role as well. Another role of protocols concerns the enabling of functionality to improve energy efficiency at the network level, such as discovery protocols that allow for quick adaptation to network components being taken dynamically into and out of service depending on network conditions.

o At the network level. Perhaps the greatest opportunities to realize power savings exist at the level of the network as whole. For example, optimizing energy efficiency may involve directing traffic in such a way that it allows for isolation of equipment that may at the moment not be needed so that it could be powered down or brought into power-saving mode. By the same token, traffic should be directed in a way that requires bringing additional equipment online or out of power-saving mode in cases where alternative traffic paths are available for which the incremental energy cost would amount to zero. Likewise, some networking devices may be more power-intensive than others whose use might be avoided unless required to meet peak capacity demands. Generally, incremental power consumption can be viewed as a cost metric that networks should strive to minimize and consider as part of routing and of network path optimization.

o At the architecture level. The current network architecture supports a wide range of applications, but does not take into account energy efficiency as one of its design parameters. One can argue that the most energy efficient shift of the last two decades has been the deployment of Content Delivery Network overlays: while these were set up to reduce latency and minimize bandwidth consumption, from a network perspective, retrieving the content from a local cache is also much greener. What other architectural shifts can produce energy consumption reduction?
We believe that network standardization organizations in general, and IETF in particular, can make important contributions to each of these vectors. In this document, we will there explore each of those vectors in further detail and for each point out specific challenges for IETF.

It should be noted that this document borrows heavily from material from a prior paper, [GreenNet22]. This material has been both expanded (for example, in terms of some of the opportunities) and pruned (for example, in terms of background on prior scholarly work). In addition, unlike the prior paper, this document focuses on and attempts to articulate specific challenges as related to work that could be championed by the IETF.

2. Definitions and Acronyms

TBD

3. Contributors to Network Energy Consumption

When exploring possibilities to improve energy efficiency, it is important to understand which aspects contribute to power consumption the most and hence where the greatest potential for power savings lies.

Power is ultimately drawn from devices. The power consumption of the device can be divided into the consumption of the core device - the backplane and CPU, if you will - as well as additional consumption incurred per port and line card. Furthermore it is important to understand the difference between power consumption when a resource is idling versus when it is under load. This helps to understand the incremental cost of additional transmission versus the initial cost of transmission.

In typical networking devices, only roughly half of the energy consumption is associated with the data plane [bolla2011energy]. An idle base system typically consumes more than half of the power over the same system running at full load [chabarek08], [cervero19]. This means that a device’s power consumption increases not linearly with the volume of forwarded traffic but resembles more of a step function. Generally, the cost of the first bit is very high, as it requires powering up a device, port, etc. The cost of transmission of additional bits (beyond the first) is many orders of magnitude lower. Likewise, the incremental cost of incremental CPU and memory needed to process additional packets becomes fairly negligible. By the same token, generally speaking it is more energy-efficient to transmit a large volume of data in one burst (and turning off the interface when idling), instead of continuously transmitting at a
lower rate. In that sense it can be the duration of the transmission that dominates the energy consumption, not the actual data rate.

The implications on green networking from an energy-savings standpoint are significant: Potentially the largest gains can be made when network resources can effectively be taken off the grid (i.e. isolated and removed from service so they can be powered down while not needed). Likewise, for applications where this is possible, it may be desirable to replace continuous traffic at low data rates with traffic that is sent in burst at high data rates, in order to potentially maximize the time during which resources can be idled.

At the same time, any non-idle resources should be utilized to the greatest extent possible as the incremental energy cost is negligible. Of course, this needs to occur while still taking other operational goals into consideration, such as protection against failures (allowing for readily-available redundancy and spare capacity in case of failure) and load balancing (for increased operational robustness). As data transmission needs tend to fluctuate wildly and occur in bursts, any optimization schemes need to be highly adaptable and allow for very short control loops.

As a result, emphasis needs to be given to technology that allows to (for example) (at the device level) exercise very efficient and rapid discovery, monitoring, and control of networking resources so that they can be dynamically be taken offline or back into service, without (at the network level) requiring extensive convergence of state across the network or recalculation of routes and other optimization problems, and (at the network equipment level) support rapid power cycle and initialization schemes.

4. Challenges and Opportunities - Equipment Level

Perhaps the most obvious opportunities to make networking technology more energy efficient exist at the equipment level. After all, networking involves physical equipment to receive and transmit data. Making such equipment more power efficient, have it dissipate less heat to consume less energy and reduce the need for cooling, making it eco-friendly to deploy, sourcing sustainable materials and facilitating recycling of equipment at the end of its life-cycle all contribute to making networks greener. More specific and unique to networking are schemes to reduce energy usage of transmission technology from wireless (antennas) to optical (lasers).

Beyond such "first-order" opportunities, network equipment just as importantly plays an important role to enable and support green networking at other levels. Of prime importance is the equipment’s ability to provide visibility to management and control plane into
its current energy usage. Such visibility enables control loops for energy optimization schemes, allowing applications to obtain feedback regarding the energy implications of their actions, from setting up paths across the network that require the least incremental amount of energy to quantifying metrics related to energy cost used to optimize forwarding decisions.

One prerequisite to such schemes is to have proper instrumentation in place that allows to monitor current power consumption at the level of networking devices as a whole, line cards, and individual ports. Such instrumentation should also allow to assess the energy efficiency and carbon footprint of the device as a whole. In addition, it would be desirable to relate this power consumption to data rates as well as to current traffic, for example, to indicate current energy consumption relative to interface speeds, as well as for incremental energy consumption that is expected for incremental traffic (to aid control schemes that aim to "shave" power off current services or to minimize the incremental use of power for additional traffic). This is an area where the current state of the art is sorely lacking and standardization lags behind; for example, as of today, no corresponding standardized YANG data models [RFC7950] for network energy consumption that can be used in conjunction with management and control protocols have been defined.

Instrumentation should also take into account the possibility of virtualization, introducing layers of indirection to assess the actual energy usage. For example, virtualized networking functions could be hosted on containers or virtual machines which are hosted on a CPU in a data center instead of a regular network appliance such as a router or a switch, leading to very different power consumption characteristics. For example, a data center CPU could be more power efficient and consume power more proportionally to actual CPU load. Instrumentation needs to reflect these facts and facilitate attributing power consumption in a correct manner.

Beyond monitoring and providing visibility into power consumption, control knobs are needed to configure energy saving policies. For instance, power saving modes are common in endpoints (such as mobile phones or notebook computers) but sorely lacking in networking equipment.

5. Challenges and Opportunities - Protocol Level

There are several opportunities for energy savings at the protocol level. We characterize them along three main categories: protocols designed to reduce the volume of data to be transmitted; protocols designed to optimize data transmission rates under energy considerations; and protocols that enable energy optimization schemes
at the network level. A fourth category, "other", is used to capture any other aspects not easily categorized into the other three.

5.1. Data Volume Reduction

The first category involves designing protocols in such a way that they reduce the volume of data that needs to be transmitted for any given purpose. Loosely speaking, by reducing this volume, more traffic can be served by the same amount of networking infrastructure, hence reducing overall energy consumption. Possibilities here include protocols that avoid unnecessary retransmissions. At the application layer, protocols may also use coding mechanisms that encode information close to the Shannon limit. Currently, most of the traffic over the Internet consists of video streaming and encoders for video are already quite efficient and keep improving all the time, resulting in energy savings as one of many advantages (of course being offset by increasingly higher resolution). However, it is not clear that the extra work to achieve higher compression ratios for the payloads results in a net energy gain: what is saved over the network may be offset by the compression/decompression effort. Further research on this aspect is necessary.

At the transport protocol layer, TCP and to some extent QUIC react to congestion by dropping packets. This is a highly energy inefficient method to signal congestion, since the network has to wait one RTT to be aware that the congestion has occurred, and since the effort to transmit the packet from the source up until it is dropped ends up being wasted. This calls for new transport protocols that react to congestion without dropping packets. ECN[RFC2481] is a possible solution, however not widely deployed. DC-TCP [alizadeh2010DCTCP] is tuned for the Data Center. Qualitative Communication [QUAL][westphal2021qualitative] allows the nodes to react to congestion by dropping only some of the data in the packet, thereby only partially wasting the resource consumed by transmitted the packet up to this point. We believe there is a need for novel transport protocols for the WAN that ensures that no energy is wasted transmitting packets that will be eventually dropped.

Another solution to reduce the bandwidth of network protocols by reducing their header tax, for example applying header compression. An example in IETF is [RFC3095]. Again, reducing protocol header size saves energy to forward packets, but at the cost of maintaining a state for compression/decompression, plus computing these operations. The gain from such protocol optimization further depends on the application and whether it sends packets with large payloads close to the MTU (the header tax and any savings here are very
limited), or whether it sends packets with very small payload size (making the header tax more pronounced and savings more significant).

An alternative to reducing the amount of protocol data is to design routing protocols that are more efficient to process at each node. For instance, path based forwarding/labels such as MPLS [RFC3031] facilitate the next hop look-up, thereby reducing the energy consumption. It is unclear if some state at router to speed up look up is more energy efficient that "no state + lookup" that is more computationally intensive. Other methods to speed up a next-hop lookup include geographic routing (e.g. [herzen2011PIE]). Some network protocols could be designed to reduce the next hop look-up computation at a router. It is unclear if Longest Prefix Match (LPM) is inefficient from an energy point of view, or if it is a significant energy budget cost for the operation of a router.

5.2. Traffic Adaptation

The second category involves designing protocols in such a way that the rate of transmission is chosen to maximize energy efficiency. For example, Traffic Engineering (TE) can be manipulated to impact the rate adaptation mechanism [ren2018jordan]. By choosing where to send the traffic, TE can artificially congest links so as to trigger rate adaptation and therefore reduce the total amount of traffic. Most TE systems attempt to minimize Maximal Link Utilization (MLU) but energy saving mechanisms could decide to do the opposite (maximize minimal link utilization) and attempt to turn off some resources to save power.

5.3. Enabling Network Energy Saving Mechanisms

Novel protocols are also needed in two dimensions: to discover what links are available and/or energy efficient. For instance, links may be turned off in order to save energy, and turned back on based upon the elasticity of the demand. Protocols should be devised to discover when this happens, and to have a view of the topology that is consistent with frequent topology updates due to power cycling of the network resources.

Also, protocols are required to quickly converge onto an energy-efficient path once a new topology is created by turning links on/off. Current routing protocols may provide for fast recovery in the case of failure. However, failures are hopefully relatively rare events, while we expect an energy efficient network to aggressively try to turn off links.

Some mechanism is needed to present to the management layer a view of the network that identifies opportunities to turn resources off.
(routers/links) while still providing some decent level of Quality of Experience (QoE) to the users. This gets more complex as the level of QoE shifts from the current Best Effort delivery model to more sophisticated mechanisms with, for instance, latency, bandwidth or reliability guarantees.

5.4. Network Addressing

There are other ways to shave off energy usage from networks. One example concerns network addressing. Address tables can get very large, resulting in large forwarding tables that require considerable amount of memory, in addition to large amounts of state needing to be maintained and synchronized. From an energy footprint perspective, both can be considered wasteful and offer opportunities for improvement. At the protocol level, rethinking how addresses are structured can allow for flexible addressing schemes that can be exploited in network deployments that are less energy-intensive by design. This can be complemented by supporting clever address allocation schemes that minimize the number of required forwarding entries as part of deployments.

6. Challenges and Opportunities - Network Level

Networks have been optimized for many years under many criteria, for example to optimize (maximize) network utilization and to optimize (minimize) cost. Hence, it is straightforward to add optimization for "greenness" (including energy efficiency, power consumption, carbon footprint) as important criteria.

This includes assessing the carbon footprints of paths and optimizing those paths so that overall footprint is minimized, then applying techniques such as path-aware networking or segment routing [RFC8402] to steer traffic along those paths. It also includes aspects such as considering the incremental energy usage in routing decisions. Optimizing cost has a long tradition in networking; many of the existing mechanisms can be leveraged for greener networking simply by introducing energy footprint as a cost factor. Low-hanging fruit include the inclusion of energy-related parameters as a cost parameter in control planes, whether distributed (e.g. IGP) or conceptually centralized via SDN controllers.

Other opportunities concern adding energy-awareness to dynamic path selection schemes, requiring corresponding instrumentation as mentioned earlier. Again, considerable energy savings can potentially be realized by taking resources offline (e.g. putting them into power-saving or hibernation mode) when they are not currently needed under current network demand and load conditions. Therefore, weaning such resources from traffic becomes an important
consideration for energy-efficient traffic steering. This contrasts and indeed conflicts with existing schemes that typically aim to create redundancy and load-balance traffic across a network to achieve even resource utilization. This usually occurs for important reasons, such as making networks more resilient, optimizing service levels, and increasing fairness. One of the big challenges hence concerns how resource weaning schemes to realize energy savings can be accommodated while preventing the cannibalization of other important goals, counteracting other established mechanisms, and avoiding destabilization of the network.

As an important prerequisite to capture many of those opportunities, good abstractions (and corresponding instrumentation) that allow to easily assess energy cost and carbon footprint will be required. These abstractions need to account for not only the energy cost associated with packet forwarding across a given path, but related cost for processing, for memory, for maintaining of state, to result in a holistic picture. Optimization of carbon footprint involves in many cases trade-offs that involve not only packet forwarding but also aspects such as keeping state, caching data, or running computations at the edge instead of elsewhere. (Note: there may be a differential in running a computation at an edge server vs. at an hyperscale DC. The latter is often better optimized than the latter.) Likewise, other aspects of carbon footprint beyond mere energy-intensity should be considered. For instance, some network segments may be powered by more sustainable energy sources than others, and some network equipment may be more environmentally-friendly to build, deploy and recycle, all of which can be reflected in abstractions to consider.

A related set of challenges concerns the fact that such schemes result in much greater dynamicity and continuous change in the network as resources may be getting steered away from (when possible) and then leveraged again (when necessary) in rapid succession. This imposes significant stress on convergence schemes that results in challenges to the scalability of solutions and their ability to perform in a fast-enough manner. Network-wide convergence imposes high cost and incurs significant delay and is hence not susceptible to such schemes. The impact will in all likelihood needs to be mechanisms that do not require convergence beyond the vicinity of the affected network device. Especially in cases where central network controllers are involved that are responsible for aspects such as configuration of paths and the positioning of network functions and that aim for global optimization, the impact of churn needs to be minimized. This means that, for example, extensive recalculation e.g. of routes and paths based on the current energy state of the network needs to be avoided.
An opportunity may lie in making a distinction between "energy modes" of different domains. For instance, in a highly trafficked core, the energy challenge is to transmit the traffic efficiently. The amount of traffic is relatively fluid (due to multiplexing of multiple sessions) and the traffic is predictable. In this case, there is no need to optimize on a per session basis nor even at a short time scale. In the access networks connecting to that core, though, there are opportunities for this fast convergence: traffic is much more bursty, less predictable and the network should be able to be more reactive. Other domains such as DCs may have also more variable workloads and different traffic patterns.

7. Challenges and Opportunities - Architecture Level

Another possibility to improve network energy efficiency is to organize networks in a way that they can best serve important applications so as to minimize energy consumption. Examples include retrieval of content or remote computation. This allows to minimize the amount of communication that needs to take place in the first place, although energy savings within the network may at least in part be offset by additional energy consumption elsewhere. The following are some examples that suggest that it may be worthwhile reconsidering the ways in which networks are architected to minimize their carbon footprint.

For example, Content Delivery Networks (CDNs) have reduced the energy expenditure of the Internet by downloading content near the users. The content is sent only a few times over the WAN, and then is served locally. This shifts the energy consumption from networking to storage. Further methods can reduce the energy usage even more [bianco2016energy] [mathew2011energy] [islam2012evaluating]. Whether overall energy savings are net positive depends on the actual deployment, but from the network operator's perspective, at least it shifts the energy bill away from the network to the CDN operator.

While CDNs operate as an overlay, another architecture has been proposed to provide the CDN features directly in the network, namely Information Centric Networks [ahlgren2012survey], studied as well in the IRTF ICNRG. This however shifts the energy consumption back to the network operator and requires some power-hungry hardware, such as chips for larger name look-ups and memory for the in-network cache. As a result, it is unclear if there is an actual energy gain from the dissemination and retrieval of content within in-network caches.

Fog computing and placing intelligence at the edge are other architectural directions for reducing the amount of energy that is spent on packet forwarding and in the network. There again, the trade-off is between performing computation in a an energy-optimized
data center at very large scale, but requiring transmission of significant volumes of data across many nodes and long distances, versus performing computational tasks at the edge where the energy may not be used as efficiently (less multiplexing of resources, and smaller sites are inherently less efficient due to their smaller scale) but the amount of long-distance network traffic is significantly reduced. Softwarization, containers, microservices are direct enablers for such architectures, and the deployment of programmable network infrastructure (as for instance Infrastructure Processing Units - IPUs or smartNICs that offload some computations from the CPU onto the NIC) will help its realization. However, the power consumption characteristics of CPUs are different from those of NPUs, another aspect to be considered in conjunction with virtualization.

Other possibilities concern taking economic aspects into consideration impact, such as providing incentives to users of networking services in order to minimize energy consumption and emission impact. An example for this is given in [wolf2014choicenet], which could be expanded to include energy incentives.

Other approaches consider performing a late binding of data and functions to be performed on the data [krol2017NFaaS]. The COIN Research Group in IRTF focuses on similar issues. Jointly optimizing for the total energy cost, taking into account networking and computing (and the different energy cost of computing in an hyperscale DC vs an edge node) is still an area of open research.

In summary, rethinking of the overall network (and networked application) architecture can be an opportunity to significantly reduce the energy cost at the network layer, for example by performing tasks that involve massive communications closer to the user. To what extend these shifts result in a net reduction of carbon footprint is an important question that requires further analysis on a case-by-case basis.

8. Conclusions

How to make networks "greener" and reduce their carbon footprint is an important problem for the networking industry to address, both for societal and for economic reasons. This document has highlighted some of the technical challenges and opportunities in that regard, for example:

- Equipment instrumentation advances for improved energy-awareness, definition and standardization of granular management information;
- Protocol advances for improving the ratio of goodput to throughput and to reduce waste: reduction in header tax, in protocol verbosity, improvements in coding, etc.

- Protocol advances to enable rapidly taking down, bring back online, and discover availability and power saving status of networking resources while minimizing the need for reconvergence and propagation of state;

- Network advances to allow to dynamically take resources offline where feasible while minimizing churn;

- Energy footprint aware traffic steering and routing; carbon footprint as a traffic cost metric to optimize;

- Reorganization of networking architecture for important classes of applications (examples: content delivery, right-placing of computational intelligence) to optimize green foot print and holistic approaches to trade off carbon footprint between forwarding, storage, and computation;

- Security issues imposed by greater energy awareness, to minimize the new attack surfaces that would allow an adversary to turn off resources, or to waste energy;

- Reliability issues for a network that relies on fewer resource diversity, and with more operational complexity.

Of those, perhaps the key challenge to address right away concerns the ability to expose at a fine granularity the energy impact of any networking actions. Providing visibility into this will enable many approaches to come towards a solution. It will be key to implementing optimization via control loops that allow to assess the energy impact of decision taken. It will also help to answer questions such as: is caching - with the associated storage energy - better than retransmitting from a different server - with the associated networking cost? Is compression more energy-efficient once factoring the computation cost of compression vs transmitting uncompressed data? Which compression scheme is more energy efficient? Is energy saving of computing at an efficient hyperscale DC compensated by the networking cost to reach that DC? Is the overhead of gathering and transmitting fine-grained energy telemetry data offset by the total energy gain by ways of better decisions that this data enables? Is transmitting data to a LEO constellation compensated by the fact that once in the constellation, the networking is fueled on solar energy? Is the energy cost of sending rockets to place routers in Low Earth Orbit amortized over time?
Determining where the sweet spots are and optimizing networks along those lines will be a key towards making networks "greener". We expect to see significant advances across these areas and believe that IETF has an important role to play in facilitating this.

9. IANA Considerations

This document does not have any IANA requests.

10. Security Considerations

Security considerations may appear to be orthogonal to green networking considerations. However, there are a number of important caveats.

Security vulnerabilities of networks may manifest themselves in compromised energy efficiency. For example, attackers could aim at increasing energy consumption in order to drive up attack victims' energy bill. Specific vulnerabilities will depend on the particular mechanisms. For example, in the case of monitoring energy consumption data, tampering with such data might result in compromised energy optimization control loops. Hence any mechanisms to instrument and monitor the network for such data need to be properly secured to ensure authenticity.

In some cases there are inherent tradeoffs between security and maximal energy efficiency that might otherwise be achieved. An example is encryption, which requires additional computation for encryption and decryption activities and security handshakes, in addition to the need to send more traffic than necessitated by the entropy of the actual data stream. Likewise, mechanisms that allow to turn resources on or off could become a target for attackers.

11. Acknowledgments

Acknowledgments will be added at a later stage.

12. Informative References (TBD)

[ahlgren2012survey]


[telefonica2020]

[westphal2021qualitative]

[wolf2014choicenet]

Authors’ Addresses

Alexander Clemm
Futurewei
2330 Central Expressway
Santa Clara, CA 95050
USA

Email: ludwig@clemm.org

Cedric Westphal
Futurewei

Email: cedric.westphal@futurewei.com

Jeff Tantsura
Microsoft

Email: jefftant.ietf@gmail.com

Laurent Ciavaglia
Rakuten Mobile

Email: laurent.ciavaglia@rakuten.com

Marie-Paule Odini

Email: mp.odini@orange.fr
Abstract

This memo provides an overview of work performed by or proposed within the IETF related to energy and/or green: awareness, management, control or reduction of consumption of energy, and sustainability as it related to the IETF.

This document is written to help those unfamiliar with the work but interested in it, in the hope to raise more interest in energy-related activities within the IETF, such as identifying gaps and investigating solutions as appropriate.

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 12 January 2023.

Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.
Table of Contents

1. Introduction .................................................. 3
2. Energy Saving: An Introduction ............................... 3
   2.1. Digitization ............................................. 4
   2.2. Energy Saving Through Scale ............................. 4
      2.2.1. An Example: Telephony ............................. 5
      2.2.2. The Packet Multiplexing Principle .................... 5
      2.2.3. End-to-End Transport ............................... 5
      2.2.4. Global vs Restricted Connectivity: The Internet Routing Architectures ....................... 5
      2.2.5. Freedom to Innovate .................................. 6
      2.2.6. End-to-End Encryption ............................... 6
      2.2.7. Converged Networks ................................... 6
         2.2.7.1. IntServ and DetNet .............................. 7
         2.2.7.2. DiffServ ......................................... 7
         2.2.7.3. SIP .............................................. 7
2. Higher or New Energy Consumption ............................ 8
4. Some Notes on Sustainability .................................. 9
   4.1. Follow the Energy Cloud Scheduling ...................... 9
   4.2. Minimize Generated Heat ................................ 10
   4.3. Heat Recovery .......................................... 10
   4.4. Telecollaboration ....................................... 10
5. Energy Optimization in Specific Networks ................... 12
   5.1. Low Power and Lossy Networks (LLN) ..................... 12
      5.1.1. 6LOWPAN WG ........................................ 13
      5.1.2. LPWAN WG .......................................... 13
      5.1.3. 6TISCH WG ......................................... 13
      5.1.4. 6LO WG ........................................... 14
      5.1.5. ROLL WG ........................................... 14
   5.2. Constrained Nodes and Networks .......................... 15
      5.2.1. LWIG WG ........................................... 15
      5.2.2. CoRE and CoAP ...................................... 15
      5.2.3. Satellite Constellations ............................ 16
      5.2.4. Devices with Batteries .............................. 16
   5.3. Sample Technical Enablers ................................ 17
      5.3.1. (IP) Multicast ....................................... 17
         5.3.1.1. Power Saving with Multicast ....................... 17
         5.3.1.2. Power Waste Through Multicast-based Service Coordination ............................... 18
         5.3.1.3. Multicast Problems in Wireless Networks .... 18
      5.3.2. Sleepy Nodes ........................................ 19
This document summarizes work that has been proposed to or performed within the IETF/IRTF. Particularly, it covers IETF/IRTF RFCs as well as ISE RFCs and IETF/IRTF or individual submission drafts that were abandoned for various reasons (e.g., lack of momentum, broad scope).

There are various aspects how a given work can relate to energy that are classified into categories. Such a classification does not attempt to propose a formal taxonomy but it is used for the sake of better readability. Technologies are listed under a category that is specifically significant, for example, by being most narrow.

This memo usually refers to the technologies by significant early RFC or specific draft version, as opposed to the newest. This is contrary to the common practice in IETF documents to refer to the newest version. This is done because it allows readers to better understand the historic timeline in which a specific technology was introduced. Especially successful IETF technologies will have newer RFC that updates such initial work.

2. Energy Saving: An Introduction

Technologies that simply save energy compared to earlier/other alternatives are the broadest and most unspecific category. In this memo such an energy saving simply refers to energy savings in some unit of electricity, such as kWh and does not take other aspects into account. See Section 4 for more details.
2.1. Digitization

Digitization describes the transformation of processes from non or less digital with networking to more digital with computer-networking. For comparable process results, the digitized option is often, but not always, less energy consuming. Consider for example energy consumption in the evolution of messaging starting from postal mail and over telegrams and various other historic form to solutions including e-mail utilizing for example the IETF "Simple Mail Transport Protocol" (SMTP, [RFC822]), group communications utilizing the IETF "Network News Transport Protocol" (NNTP, [RFC3977]) or the almost infinite set of communication options built on top of the IETF "HyperText Transport Protocol" (HTTP, [RFC2086] and successors) and IETF "HyperText Markup Language" (HTML, [RFC1866] and successors).

Traditionally, digitization had only "incidental", but not "intentional" relationship to energy consumption: If it saved energy, this was not only not a target benefit, it was not even recognized as one, until probably recently. Instead, the evolution was driven from anything-but-energy benefits, but instead utility benefits such as improved speed, functionality/flexibility, accessibility, scalability and reduced cost.

In hindsight though, digitization through IETF technologies and specifically the Internet will likely have the largest contribution to energy saving amongst all the possible categories, but it is also the hardest to pinpoint on any specific technology/RFC. Instead, it is often a combination of the whole stack of deployed protocols and operational practices that contributes to energy saving through digitization. It is likely also the biggest overall energy saving impact of all possible categories that relate IETF work to energy:

The Internet as well as all other TCP/IP networks are likely the biggest energy saving development of the past few decades if only the energy consumption of equivalent services is compared. On the other hand, they are also the cause for the biggest new type of new energy consumption because of all the new services introduced in the past decades with the Internet and the hyper-scaling that the Internet affords them.

2.2. Energy Saving Through Scale
2.2.1. An Example: Telephony

In most cases, energy saving through the use of IETF protocols compared to earlier (digitized or non digitized) solutions is purely a result of the reduction in the energy cost per bit over the decades in networking. For example, the energy consumption of digital voice telephony through the IETF "Session Initiation Protocol" (SIP, [RFC2543] and successors) can easily be assumed to be more energy efficient on a per voice-minute basis than prior voice technologies such as analog or digital "Time Division Multiplex" (TDM) telephony solely because of this evolution in mostly device as well as physical-layer and link-layer networking technologies.

2.2.2. The Packet Multiplexing Principle

Nevertheless, it is at the heart of the packet multiplexing model employed by the IETF networking protocols IP ([RFC791]) and IPv6 ([RFC1883] and successors) to successfully support this scaling that brough down the cost per bit through ever faster links and network nodes, especially for networks larger than building scale networks. While the IETF protocols have not been the first or over their early decades necessarily the most widely deployed packet networking protocols, they where the ones who at least during the 1990th started to break away from other protocols both in scale of deployment, as well as in development of further technologies to support this scaling.

2.2.3. End-to-End Transport

At the core of scalability, even up to now, is the lightweight per-packet-processing enabled through end-to-end congestion loss management architecture as embodied through the IETF "Transmission Control Protocol" (TCP, [RFC793] and successors, e.g. [I-D.ietf-tcpm-rfc793bis]). This model eliminated more expensive per-hop, per-packet processing, such as would be required for reliable hop-by-hop forwarding through per-hop ARQ, which was key to scaling routers cost effectively.

2.2.4. Global vs Restricted Connectivity: The Internet Routing Architectures

The meshed peer-to-peer and transitive routing of the Internet enabled through the IETF "Border Gateway (Routing) Protocol (BGP, [RFC4271] as well as predecessors) is another key factor to successful scalability, because it enabled competitive market forces to explore markets quickly.
Prior to the Internet, the public often only had access to highly regulated international networking connections through often per-country monopoly regulated data networks.

2.2.5. Freedom to Innovate

(non-IP) networks often also did not allow as much "freedom-to-innovate" (as it is often called in the IETF) for applications running over it. Instead those networks where exploring the coupling of packet transport with higher layer services to allow the network operator some degree of revenue sharing with the services running on top of it. Such approaches resulted not only in higher cost of those services but also (likely) preferential and (often) exclusionary treatment of network traffic not fitting the perceived highest revenue service options.

2.2.6. End-to-End Encryption

When the same business practices where applied to IP network, it was one of the key factors leading to the development of IETF end-to-end encryption though protocols such as "Transport Layer Security" (TLS, [RFC2246] and its successors). This further strengthened the ability to scale service/applications at minimum additional cost for the underlying packet transport, arguably driving innovation into ever faster networking technology and likely lower cost per bit.

2.2.7. Converged Networks

Another key factor to support scaling where IETF technologies that allowed to multiplex different types of traffic (e.g., realtime vs. non-realtime) which previously used separate networks with typically incompatible networking technologies.

Eliminating multiple physical networks with separate routing/forwarding nodes and separate links affords significant energy savings even at the same generation of speed and hence energy/bit simply by avoiding the N-fold production and operations of equipment and links. Of course, originally the CAPEX and OPEX of multiple, technology-diverse networks and host-stacks was the core reason for unified networks, and energy saving is in hindsight just incidental (as for all other cases mentioned here).
2.2.7.1. IntServ and DetNet

The first (non-IETF) wider adopted technology promising converged networks was "Asynchronous Transfer Mode" (ATM), which was designed and deployed at the end of the 1980th to support specifically multiplexing of "Data Voice and Video", where both Voice and Video (at that time) required loss-free deterministic bounded latency and low-jitter and had therefore their own Time-Division-Multiplex (TDM) networks, both separate from so-called Data networks using packet multiplexing. This technology was very expensive on a per-bit basis due to its cell-forwarding nature though.

At the end of the 1980th, it was proven in [BOUNDED_LATENCY] that variable length packet multiplexing in network can also support non-NP-hard calculations for bounded latency. This lead to the IETF "Integrated Services WG" (INTSERV) to support such guaranteed throughput and bounded latency traffic via [RFC2212] – and to the demise of ATM.

IntServ has so far seen little traction because it too got superceeded as explained in the following section – for its original use-cases (voice and video). However this type of services are being revisited for a broader set of use-cases [RFC8575] in the DetNet WG, which should enable even further network infrastructure convergence for IoT and industrial markets.

2.2.7.2. DiffServ

Due to the much higher per-packet processing overhead of INSERV versus standard (so-called Best-Effort) Internet traffic, the INTSERV model was already recognized in the 1990th to not support highest-scale at lowest cost, leading to the parallel development of the IETF "Differentiated Services WG" (DIFFSERV) model defined in [RFC2475]. This has since then become the dominant technology to support multiplexing of applications and services originally not designed for the Internet onto a common TCP/IP network infrastructure, specifically for voice and video over UDP ([RFC768]) including RTP ([RFC3550]) and SIP.

2.2.7.3. SIP

SIP has most notably in the past two decades eliminated additional network infrastructures previously required for (voice) telephony services starting in the early 2000 with commercial/enterprise deployments and today by removing even the option for any (non-IP/SIP) analog or digital (ISDN) telephone service connection, instead delivering those purely as services over adaptation interfaces on home routers (TBD: Any RFC to cite for those tunneling/adaptation
3. Higher or New Energy Consumption

Digitized, network centric workflows may consume more energy than their non-digitized counterpart, as may new network centric workflows without easy to compare prior workflows.

In one type of instances, the energy consumption on a per-instance basis is lower than in the non-digitized/non-Internet-digitized case, but the total number of instances that are (Internet)-digitized is orders of magnitudes larger than their alternative options, typically because of their higher utility or lower overall cost.

For example, each instance of (simple text) email consumes less energy than sending a letter or postcard. Even streaming a movie or TV series consumes less energy than renting a DVD (https://www.smithsonianmag.com/science-nature/streaming-movie-less-energy-dvd-180951586). Nevertheless, the total amount of instances and in result energy consumption for email and streaming easily outranks their predecessor technologies.

While these instances look beneficial from a simple energy consumption metric, its overall scale and the resulting energy consumption may in itself become an issue, especially when the energy demand it creates risks to outstrip the possible energy production, short term or long term. This concern is nowadays often raised against the "digital economy", where the network energy consumption is typically cited as a small contributor relative to its applications, such as what is running in Data Centers (DC).

In other cases, the energy consumption of digitization requires often significantly more than their pre-digitization alternatives. The most well-known example of this are likely crypto-currencies based on "proof-of-work" computations (mining), which on a per currency value unit can cost 10..30 times or more of the energy consumed by for example gold mining (very much depending on the highly fluctuating price of the crypto-currency). Nevertheless, its overall utility compared to such prior currencies or valuables makes it highly successful in the market.

In general, the digital economy tends to be more energy intensive on a per utility/value unit, for example by replacing a lot of manual labor with computation), and/or it allows for faster growth of its workflows.
The lower the cost of network traffic, and the more easily accessible everywhere network connectivity is, the more competitive and/or successful most of these new workflows of the digital economy can be.

Given how TCP/IP based networks, especially the Internet have excelled through their design principles (and success) in this reduction of network traffic cost and ubiquitous access over the past few decades, as outlined above, one can say that IETF technologies and especially the Internet are the most important enabler of the digital economy, and the energy consumption it produces.

4. Some Notes on Sustainability

Sustainability is the principle to utilize resources in a way that they do not diminish or run out over the long term. Beyond the above covered energy saving, sustainability relates with respect to the IETF specifically to the use of renewable sources of energy to minimize exhaustion of fossil resources, and the impact of IETF technologies on global warming to avoid worsening living conditions on the planet.

While there seems to be no IETF work specifically intending to target sustainability (TBD: did we miss anything ?), the Internet itself can similarly to how it does for digitization play a key role in building sustainable networked IT infrastructures. The following subsections list three examples areas where global high performance, low-cost Internet networking is a key requirement.

4.1. Follow the Energy Cloud Scheduling

Renewable energy resources (except for water) do commonly have fluctuating energy output. For example, solar energy output correlates to night/day and strength of sunlight. Cloud Data Centers (DC) consume a significant amount of the IT sectors energy. Some workloads may simply be scheduled to consume energy in accordance with the amount of available renewable energy at the time, not requiring the network. Significant workloads are not elastic in time, such as interactive cloud DC interactive work (cloud based applications) or entertainment (gaming, etc.). These workloads may be instantiated or even dynamically (over time) migrate to a DC location with sufficient renewable energy and the Internet (or large TCP/IP OTT backbone networks) will serve as the fabric to access the remote DC and to coordinate the instantiation/migration.
4.2. Minimize Generated Heat

The majority of energy in cloud DC is normally also wasted as exhaust heat, requiring even more energy for cooling. The warmer the location, the more energy needs to be spent for cooling. For this reason, DC in cooler climates such as https://greenmountain.no/power-and-cooling/ can help to reduce the overall DC energy consumption significantly (independent of the energy being consumed in the DC to be renewable itself). The Internet again plays the role of providing access to those type of DC whole location is not optimized for consumption but for sustainable generation of compute and storage.

4.3. Heat Recovery

Exhaust heat, especially from compute in DC, can be recovered when it is coupled to heating systems ranging in size all the way from individual familys home through larger buildings (hotels etc.) all the way to district heating systems. A provider of such type of compute-generated heat as a service can sell the compute capacity as long as there is cost efficient network connectivity. "Cloud & Heat" is an example company offering such infrastructures and services https://www.cloudandheat.com/wp-content/uploads/2020/02/2020_CloudHeat-Whitepaper-Cost-saving-Potential.pdf.

4.4. Telecollaboration

Telecollaboration has a long history in the IETF resulting in multiple core technologies over the decades.

If one considers textual communications via email and netwnews (using e.g.: NNTP) as early forms of Telecollaboration, then telecollaboration history through IETF technology reaches back into the 1980th and earlier.

Around 1990, the IETF work on IP Multicast (e.g.[RFC1112] and later) enabled the first efficient forms of audio/video group collaboration through an overlay network over the Internet called the MBone https://en.wikipedia.org/wiki/Mbone which was also used by the IETF for more than a decade to provide remote collaboration for its own (in-person + remote participation) meetings.
With the advent of SIP in the early 2000, commercial telecollaboration started to be built most often on SIP based session and application protocols with multiple IETF working groups contributing to that protocol suite (TBD: how much more example/details should we have here). Using this technology and the Internet, the immersive nature of telecollaboration was brought to life-size video, was/is called Telepresence https://en.wikipedia.org/wiki/Telepresence and later to even more immersive forms such as AR/VR telecollaboration.

In 2011, the IETF opened the "Real-Time Communication in WEB-browsers" (RTCWEB) WG, that towards the end of that decade became the most widely supported cross-platform standard for hundreds of commercial and free tele-collaboration solutions, including Cisco Webex, which is also used by the IETF itself, Zoom and the new IETF collaboration suite MeetEcho (TBD: good references here ?).

While the various forms of Telecollaboration are mostly instances of digitization, they are discussed under sustainability because of its comparison to in-person travel that is not based on simple comparison of energy, but nowadays by comparing their impact on global warming, a key factor to sustainability.

Telecollaboration was primarily developed because of the utility for the participants - to avoid travel for originally predominantly business communications/collaborations. It saw an extreme increase in use (TBD: references) in the Corona Crisis of 2019, when especially international travel was often prohibited, and often even working from an office. This forced millions of people to work from home and utilizing commercial telecollaboration tools. It equally caused most in-person events that where not cancelled to be moved to a telecollaboration platform over the Internet - most of them likely relying on RTCWEB protocols.

Actual energy consumption related comparison between teleconferencing and in-person travel is complex but since the last decades is commonly based on calculating some form of CO2 emission equivalent of the energy consumed, hence comparing not simply the energy consumption, but weighing it by the impact the energy consumption has on one of the key factors (CO2 emission) known to impact sustainable living conditions.

[VC2014] is a good example of a comparison between travel and telecollaboration taking various factors into account and using CO2 emission equivalents as its core metric. That paper concludes that carbon/ energy cost of telecollaboration could be as little as 7% of an in-person meeting. in-person meeting. Those numbers have various assumptions and change when time-effort of participants is converted
to carbon/energy costs. These numbers should even be better today in favor of telecollaboration: cost of Internet traffic/bit goes down while cost of fossil fuel for travel goes up.

Recently, air travel has also come under more scrutiny because the greenhouse gas emissions of air travel at the altitudes used by commercial aviation have been calculated to have a higher global warming impact than simply the amount of CO2 used by the air plane if it was exhausted at surface level. One publicly funded organization offering carbon offset services calculates a factor 3 of the CO2 consumption of an air plane https://www.atmosfair.de/de/fliegen_und_klima/flugverkehr_und_klima/klimawirkung_flugverkehr/.

In summary: Telecollaboration has a higher sustainability benefit compared to travel than just the comparison of energy consumption because of the higher challenge to use renewable energy in transportation than in networking, and this is most extreme in the case of telecollaboration that replaces air travel because of the even higher global warming impact of using fossil fuels in air travel.

5. Energy Optimization in Specific Networks

5.1. Low Power and Lossy Networks (LLN)

Low Power and Lossy Networks are networks in which nodes and/or radio links have constraints. Low power consumption constraints in nodes often originate from the need to operate nodes from as long as possible from battery and/or energy harvesting such as (today most commonly) solar panels associated with the node or ambient energy such as energy harvesting from movement for wearable nodes or piezo cells to generate energy for mechanically operated nodes such as switches.

Several IETF WGs have or are producing work primarily intended to support LLN through multiple layers of the protocol stack. [RFC8352] gives a good overview of the energy consumption related communication challenges and solutions produced by the IETF for this space.
To minimize the energy needs for such nodes, their network data-processing mechanisms have to be optimized. This includes packet header compression, fragmentation (to avoid latency through large packets at low bitrates), packet bundling to only consume radio energy at short time periods, radio energy tuning to just reach the destination(s), minimization of multicasting to eliminate need of radio receivers to consume energy and so on. [RFC8352] gives a more detailed overview, especially because different L2 technologies such as IEEE 802.15.4 type (low power) wireless networks, Bluetooth Low Energy (BLE), WiFi (IEEE 802.11) and DEC ULE.

In the Internet area of the IETF, several LLN specific WGs exist(ed):

5.1.1. 6LOWPAN WG

The "IPv6 over Low power WPAN (Wireless Personal Area Networks)" (6lowpan) WG ran from 2005 to 2014 and produced 6 RFC that adopt IPv6 to IEEE 802.15.4 type (low power) wireless networks by transmission procedures [RFC4949], compression of IPv6 (and transport) packet headers [RFC6282], modifications for neighbor discovery (ND) [RFC6775], as well as 3 informational RFCs about the WPAN space and applying IPv6 to it. "Transmission of IPv6 Packets over IEEE 802.15.4 Networks" [RFC4944], "Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks" [RFC6282], "Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)" [RFC6775] (6LOWPAN-ND).

5.1.2. LPWAN WG

Since 2014, the "IPv6 over Low Power Wide-Area Networks" (LPWAN) WG has produced 4 RFC for low-power wide area networks, such as LoRaWAN https://en.wikipedia.org/wiki/LoRa, with three standards, [RFC8724], [RFC8824], [RFC9011].

5.1.3. 6TISCH WG

Since 2013, the "IPv6 over the TSCH mode of IEEE 802.15.4e" (6tisch) WG has produced 7 RFC for a version of 802.15.4 called the "Time-Slotted Channel Hopping Mode" (TSCH), which supports deterministic latency and lower energy consumption through the use of scheduling traffic into well defined time slots, thereby also optimizing/minimizing energy consumption when compared to 802.15.4 without TSCH.
5.1.4. 6Lo WG

Since 2013, the "IPv6 over Networks of Resource-constrained Nodes" (6lo) WG has generalized the work of 6lowpan for LLN in general, producing 17 RFC for IPv6-over-l2foo adaptation layer specifications, information models, cross-adaptation layer specification (such as header specifications) and maintenance and informational documents for other pre-existing IETF work in this space.

5.1.5. ROLL WG

In the RouTinG (RTG) area of the IETF, the "Routing Over Low power and Lossy networks" (ROLL) WG has produced since 2008 23 RFC. Initially it produced requirement RFCs of different type of "Low-power and Lossy Networks": urban: [RFC5548], industrial [RFC5673], home automation [RFC5826] and building automation [RFC5867].

Since then its work is mostly focused on the "IPv6 Routing Protocol for Low-Power and Lossy Networks" (RPL) [RFC6550], which is used in a wide variety of the above described IPv6 instances of LLN networks and which are discussed in two ROLL applicability statement RFCs, "Applicability Statement: The Use of the Routing Protocol for Low-Power and Lossy Networks (RPL) Protocol Suite in Home Automation and Building Control" [RFC7733] and "Applicability Statement for the Routing Protocol for Low-Power and Lossy Networks (RPL) in Advanced Metering Infrastructure (AMI) Networks" [RFC8036].

The ROLL WG also wrote a more generic RFC for LLN, "Terms Used in Routing for Low-Power and Lossy Networks" [RFC7102]. RPL has a highly configurable set of functions to support (energy) constrained networks. Unconstrained root node(s), typically edge routers between the RPL network and a backbone network calculate "Destination-Oriented Directed Acyclic Graphs" (DODAG) and can use strict hop-by-hop source routing with dedicated IPv6 routing headers [RFC9008] to minimize constrained nodes routing related compute and memory requirements. "The Trickle Algorithm" [RFC6206] allows to minimize routing related packets through automatic lazy updates. While RPL is naturally a mesh network routing protocol, where all nodes are usually expected to be able to participate in it, RPL also supports even more lightweight leave nodes [RFC9010].
The 2013 [I-D.ajunior-energy-awareness-00] proposes the introducing of energy related parameters into RPL to support calculation/selection of most energy efficient paths. The 2017 "An energy optimization routing scheme for LLSs", [I-D.wang-roll-energy-optimization-scheme] observed that DODAGs in RPL tend to require more energy in nodes closer to the root and proposed specific optimizations to reduce this problem. Neither of these drafts proceeded in the IETF.

While original use-cases for RPL where energy and size limited networks, its design is to a large extend not scale limited. Because of this, and due to its reduced compute/memory requirements for the same size networks compared to other routing protocols, especially the so-called link-state "Interior Gateway routing Protocols" (IGP), such as most commonly used protocols ISIS [RFC1142] and OSPF [RFC2328], RPL has also proliferated into use-cases for non-constrained networks, for example to support the largest possible networks automatically, such as in [RFC8994].

5.2. Constrained Nodes and Networks

(Power) constrained nodes and/or networks exist in a much broader variety than coupled with low-power and lossy networks. For example WiFi and mobile network connections are not considered to be lossy networks, and personal mobile nodes with either connections are order of magnitude less constrained than nodes typically attached to LLN network. Therefore, broader work in the IETF than focused primarily on LLN typically uses just the term lightweight or constrained (nodes and networks).

5.2.1. LWIG WG

Since 2013, the "Light-Weight Implementation Guidance" (lwig) WG is has produced 6 informational RFC on the groups subject, much of which indirectly supports implementing power efficient network implementations via lightweight nodes/links, but it also addressed the topic explicitly including via the aforementioned [RFC8352] and [RFC9178], "Building Power-Efficient Constrained Application Protocol (CoAP) Devices for Cellular Networks".

5.2.2. CoRE and CoAP

In the APPLICATION (APP) area of the IETF, the "Constrained RESTful Environments" (core) WG has produced since 2010 21 RFC, most of them for or related to "The Constrained Application Protocol" (CoAP) [RFC6690], which can best be described as a replacement for HTTP for constrained environment, using UDP instead of TCP and DTLS instead of TLS, compact binary message formats instead of human readable textual
formats, RESTful message exchange semantic instead of a broader set of options (in HTTP), but also more functionality such as (multicast) discovery and directory services, therefore providing a more comprehensive set of common application functions with more compact on-the-wire/radio encoding than its unconstrained alternatives. "Object Security for Constrained RESTful Environments" (OSCORE), [RFC8613] is a further product of the CoRE WG providing a more message layer based, more lightweight security alternative to DTLS.

While originally designed for LLN, CoAP is transcending LLN and equally becoming standards in unconstrained environments such as wired/ethernet industrial Machine 2 Machine (M2M) communications, because of simplicity, flexibility and relying on the single set of protocols supporting the widest range of deployment scenarios.

In the SECurity (SEC) area of the IETF, the "Authentication and Authorization for Constrained Environments" (ace) working group has since 2014 produced 4 RFC for security functions in constrained environments, for example CoAP based variations of prior HTTPS protocols such as EST-coaps [RFC9148] for HTTPS based EST [RFC7030]. Constrained node support in cryptography especially entails support for Elliptic Curve (EC) public keys due to their shorter key sizes and lower compute requirements compared to RSA public keys with same cryptographic strength. While the benefits of EC over RSA where making them preferred, this "additional market space" (constrained node) benefit helped in their faster market proliferation even beyond constrained networks.

5.2.3. Satellite Constellations

Emerging communication infrastructures may have specific requirements on power consumption. Such requirements should be taken into account when designing/customizing techniques (e.g., routing) to be enabled in such networks. For example, [I-D.lhan-problems-requirements-satellite-net] identifies a set of requirements (including power) for satellite constellations.

5.2.4. Devices with Batteries

Many IETF protocols (e.g., [RFC3948]) were designed to accommodate the presence of middleboxes mainly by encouraging clients to issue frequent keepalives. Such strategy has implication on battery-supplied devices. In order to optimize battery consumption for such devices, [RFC6887] specifies a deterministic method so that client can control state in the network, including their lifetime. Keepalive alive messages may this be optimized as a function of the network policies.
A_REC#2 of [RFC7849] further insist on the importance of saving battery exacerbated by keep-alive messages and recommends the support of collaborative means to control state in the network rather than relying on heuristics.

5.3. Sample Technical Enablers

5.3.1. (IP) Multicast

5.3.1.1. Power Saving with Multicast

IP Multicast was introduced with [RFC1112] and today also called "Any Source Multicast" (ASM) has various protocols standardized in the IETF across multiple working groups. There are also MPLS and BIER multicast protocols from the IETF developed in the equally named WGs.

These three, network layer multicast technologies can be a power saving technologies when used to distribute data because they reduce the number of packets that need to be sent across the network (through in-network-replication where needed). Because most current link and router technologies do not allow to actually save significant amounts of energy on lower than maximum utilization, these benefits are often only theoretical though. Software routers are the ones most likely to expose energy consumption somewhat proportional to their throughput for just the forwarding (CPU) chip.

Likewise, in large backbone networks, IP multicast can free up bandwidth to be used for other traffic, such as unicast traffic, which may allow to avoid upgrades to faster and potentially more power consuming routers/links. Today, these benefits too are most often overcompensated for by lower per-bit energy consumption of newer generations of routers and links though.

Multicasting can also save energy on the transmitting station across radio links, compared to replicated unicast traffic, but this is rarely significant, because except for fully battery powered mesh network, there are typically non-energy-constrained nodes, such as (commonly) the wired access-points in WiFi networks.

In result, today multicasting has typically no significant power saving benefits with available network technologies. Instead it is used (for data distribution) when the amount of traffic that a unicast solution alternative (with so-called ingress replication) is not possible due to the total amount of traffic generated. This includes wireless/radio networks, where equally airtime is the limiting factor.
5.3.1.2. Power Waste Through Multicast-based Service Coordination

(IP) multicast is often not used to distribute data requested by receivers, but also coordination type functions such as service or resource announcement, discovery or selection. These multicast messages may not carry a lot of data, but they cause recurring, often periodic packets to be sent across a domain and waste energy because of various ill-advised designs, including, but not limited to the following issues:

(a) The receivers of such packets may not even need to receive them, but the protocol shares a multicast group with another protocol that the client does need to receive.

(b) The receiver should not need to receive the packet as far as multicast is concerned, but the underlying link-layer technology still makes the receiver consume the packet at link-layer.

(c) The information received is not new, but just periodically refreshed.

(d) The packet was originated for a service selection by a client, and the receiving device is even responding, but the client then chooses to select another device for the service/resource.

These problems are specifically problematic in the presence of so-called "sleepy" nodes Section 5.3.2 that need to wake up to receive such packets (unnecessarily). It is worse, when the network itself is an LLN network where the forwarders themselves are power constrained and for example periodic multicasting of such coordination packets wastes energy on those forwarders as well compared to better alternatives.

In 2006, the IETF standardized "Source Specific Multicast" (SSM) [RFC4607], a variation of IP Multicast that does not allow to perform these type of coordination functions but is only meant for (and useable for) actual data distribution. SSM was introduced for other reasons than the above-described power related issues though, but deprecating the use of ASM is one way to avoid/minimize its ill-advised use with these type of coordination functions, when energy efficiency is an issue. [RFC8815] is an example for deprecating ASM for other reasons in Service Provider networks.

5.3.1.3. Multicast Problems in Wireless Networks

[RFC9119] covers multicast challenges and solutions (proposals) for IP Multicast over Wi-Fi. With respect to power consumption, it discusses the following aspects:
(a) Unnecessary wake-up of power constrained Wi-Fi Stations (STA) nodes can be minimized by wireless Access Points (APs) that buffer multicast packets so they are sent only periodically when those nodes wake up.

(b) WiFi access points with "Multiple Input Multiple Output" (MIMO) antenna diversity focus sent packets in a way that they are not "broadcast" to all receivers within a particular maximum distance from the AP, making WiFi multicast transmission even less desirable.

(c) It lists the most widely deployed protocols using aforementioned coordination via IP multicast and describes their specific challenges and possible improvements.

(d) Existing proprietary conversion of WiFi multicast to Wi-Fi unicast packets.

[I-D.desmouceaux-ipv6-mcast-wifi-power-usage] focuses on IPv6-related concerns of multicast traffic in large wireless network. This document provides as set of statistics and the induced device power consumption of such flows.

5.3.2. Sleepy Nodes

Sleepy nodes are one of the most common design solutions in support of power saving. This includes LLN level constrained nodes, but also nodes with significant battery capacity, such as mobile phones, tablets and notebooks, because battery lifetime has long since been a key selling factor. In result, vendors do attempt to optimize power consumption across all hardware and software components of such nodes, including the interface hardware and protocols used across the nodes WiFi and mobile radios.

Restating from [I-D.bormann-core-roadmap-05]: CoAP has basic support for sleepy nodes by allowing caching of resource information in (non-sleepy) proxy nodes. [RFC7641] enhances this support by enabling sleepy nodes to update caching intermediaries on their own schedule. Around 2012/2013, there was significant review of further review of further support for sleepy nodes in CoAP, resulting in a long list of drafts, whose sleepy nodes benefits are discussed in [I-D.bormann-core-roadmap-05]: [I-D.vial-core-mirror-server], [I-D.vial-core-mirror-proxy], [I-D.fossati-core-publish-option], [I-D.giacomin-core-sleepy-option], [I-D.castellani-core-alive], [I-D.rahman-core-sleepy-problem-statement], [I-D.rahman-core-sleepy], [I-D.rahman-core-sleepy-nodes-do-we-need], [I-D.fossati-core-monitor-option]. None of these drafts proceeded though.
One partial solution to some sleepy node issues related to their
energy consumption, especially the ones caused by the use of
multicast Section 5.3.1.2, Section 5.3.1.3 is the use of the
"Constrained RESTful Environments (CoRE) Resource Directory" (CoRE-
RD) [RFC9176]. It allows for sleepy nodes to register discover and
register resources via unicast and avoids waking up sleepy nodes when
they are not selected by a resource consumer.

An partial alternative to CoRE-RD is the "DNS-Based Service
Discovery" (DNS-SD) [RFC6763] combined with for example "Service
Registration Protocol for DNS-Based Service Discovery"
[I-D.ietf-dnssd-srp]. Services can be seen as a subset of resources,
and in networks where DNS has to be supported anyhow for other
reasons, DNS-SD may be a sufficient alternative to CoRE-RD. It is
used for example in Thread https://en.wikipedia.org/wiki/
Thread_(network_protocol) for this purpose and the only multicast
based coordination is the one to establish network wide parameters,
such as the address(es) of DNS-SD server(s).

"Building Power-Efficient Constrained Application Protocol (CoAP)
Devices for Cellular Networks" [RFC9178] discusses sleepy devices,
especially the use of CoAP PubSub [I-D.ietf-core-coap-pubsub] as a
mechanism to build proxies for sleepy devices. "Sensor Measurement
Lists (SenML)", Standardized proxy infrastructures are best built
with standard data models, such as "Sensor Measurement Lists" (SenML)
[RFC8428] for sensors, likely the largest number of sleepy devices,
especially in LLN.

"Reducing Energy Consumption of Router Advertisements", [RFC7772]
eliminates/reduces the energy impact for sleepy nodes of the
ubiquitous IPv6 "Neighbor Discovery" (ND) protocol by giving
recommends for replacing multicast "Router Advertisement" (RA)
messages with so-called directed unicast versions, therefore not
waking up sleepy nodes (with an IP multicast RA message). This was
already allowed in ND [RFC4861], but not recommended as the default.
Note that [RFC7772] does not provide all the energy related
optimizations of ND as developed by 6LoWPAN through [RFC6775].
[I-D.chakrabarti-nordmark-energy-aware-nd] proposes generalizations
for those applications for to all IPv6 links, but was not further
pursued by the IETF so far.

5.4. (Lack of) Power Benchmarking Proposals

[I-D.petrescu-v6ops-ipv6-power-ipv4] presented some measurement
results of the power consumption when using IPv6 vs IPv4 with a focus
on mobile devices. Such measurements are not backed with formal
benchmarking methodologies so that solid and reliable references are
set to compare and interpret data.
https://www.ietf.org/proceedings/103/slides/slides-103-saag-iot-benchmarking-00 presented a benchmark example but with a focus on power cost of encryption.

6. Energy Management Networks

Use of IETF protocol networks in networks that operate power consumption and production is another broad area of digitization.

6.1. Smart Grid

"Smart Grid" is the most well-known instance of such energy management networks. According to https://en.wikipedia.org/wiki/Smart_grid, the term covers aspects mostly centered around intelligent measured and controlled consumption of energy. This includes "Advanced Metering Infrastructure" / "Smart Meters", remote controllable "distribution boards", "circuit breakers", "load control" and "smart appliances". Use cases for the "Smart Grid" include for example timed and measured operations of home devices such as washers or charging cars, when energy consumption is below average.

The 2011 "Internet Protocols for the Smart Grid" [RFC6272] is a quite comprehensive (66 page) overview of all IETF protocols considered to be necessary or beneficial for Smart Grid networks. This document was written in response to interest by the (not-yet-smart grid) community in utilizing the IETF TCP/IP technologies to evolve previously non-TCP/IP network, and the risk that unnecessary reinvention of the wheel/protocols would be done by that community instead of reusing what was already well specified by the IETF.

Most of the overview in this document is not specific to networks used for Smart Grid applications but just summarized in the document for the above described outreach and education to the community. The aspects most specific to Smart Grids is the back in 2011 still somewhat in its infancy adaptation of IPv6 network technologies to LLN networks (see Section 5.1 below): smart meters, circuit breakers, load measurement devices, car chargers and so on - all those devices would most likely be connected to the network via a low-power radio networks, which ideally would utilize IPv6 directly. Support for LLN networks with IPv6 has well improved in IETF specifications in the past decade.
6.2. Syncro Phasor Networks

Power output of multiple power plants/generators into the same power grid needs to be synchronized by power levels based on consumption and power phase (50/60Hz depending on continent) to avoid that energy created out-of-phase is not only wasted, but would actually burn out power lines or create permanent damage in power generators. When generators go out-of-sync, they have to be emergency switched off, resulting in (rolling-)blackouts, worsening the conditions beyond its likely root-cause such as a single overloaded limited region.

Syncro Phasor Networks are networks whose goal it is to support synchronization of power generators across a power grid, ultimately also permitting to build larger and more resilient power grids. "Power Measurement Units" (PMU) are their core sensing elements. Since about 2012? these networks have started to move from traditional SCADA towards more TCP/IP based networking and application technologies "to improve power system reliability and visibility through wide area measurement and control, by fostering the use and capabilities of synchrophasor technology" (www.naspi.org).

With their fast control loop reaction time and measurement requirements, they also benefit from reliable, fast propagation of PMU data as well as stricter clock synchronization than most Smart Grid applications. For example, transmission lines expand under heat that is caused by electrical load and/or environmental temperature by as much as 30% (between coldest and hottest or highest-load times), impacting the necessary phase relationship of power generation on either end (speed of light propagation speed based on effective length of contracted/expanded wire).

The length of transmission wires can be measured from data sent across the transmission lines and measuring their propagation latency with the help of accurate clock synchronization between sender and receiver(s), using for example network-based clock synchronization protocols. The IETF "Network Time Protocol version 4" (NTPv4), [RFC5905] is one option for this. The IEEE PTP protocol is often preferred though because it specifies better how measurements can be integrated at the hardware level of Ethernet interfaces, thus allowing easier to achieve higher accuracy, such as Maximum Time Interval (MTIE) errors of less than 1 msec. See for example [NASPILOCK].

The "North American Syncro Phasor Initiative" (NASPI), https://www.naspi.org is an example organization in support of synchrophasor networking. It is an ongoing project by the USA "Department of Energy" (DoE).
7.  (Limited) Energy Management for Networks

7.1.  Some Metrics

A 2010-2013 draft [I-D.manral-bmwg-power-usage], which was not adopted discussed and proposed metrics for power consumption that where intended to be used for benchmarking.

The later work in Section 7.2 referred instead to other metrics for measuring power consumption from other SDOs.

A 2011-2012 draft [I-D.jennings-energy-pricing], which was not adopted, discusses and proposes a data model to communicate time-varying cost of energy in support of enabling time-shifting of network attached or managed equipment consumption of power.

7.2.  EMAN WG

While the IETF did specify a few MIBs with aspects related to of power management, it was only with the formation of the "Energy Management" (EMAN) WG which ran from 2010 to 2015 and released 7 RFC, that the IETF produced a comprehensive set of MIB based standards for managing energy/power for network equipment and associated devices and integrated prior scattered power management related work in the IETF.

EMAN produced (solely) a set of data/information models (MIBs). It does not introduce any new protocol/stacks nor does it address "questions regarding Smart Grid, electricity producers, and distributors" (from [RFC7603]).

[I-D.claise-power-management-arch] describes the initial EMAN architecture as envisioned by some of the core contributors to the WG. It was rewritten in EMAN as the "Energy Management Framework" [RFC7326]. "Requirements for Energy Management" are defined in [RFC6988].

According to [RFC7326], "the (EMAN) framework presents a physical reference model and information model. The information model consists of an Energy Management Domain as a set of Energy Objects. Each Energy Object can be attributed with identity, classification, and context. Energy Objects can be monitored and controlled with respect to power, Power State, energy, demand, Power Attributes, and battery. Additionally, the framework models relationships and capabilities between Energy Objects."
One category of use-cases of particular interest to network equipment vendors was and is the management of "Power over Ethernet" via the EMAN framework, measuring and controlling ethernet connected devices through their PoE supplied power. Besides industrial, surveillance cameras and office equipment, such as WiFi access points and phones, PoE is also positioned as a new approach for replacing most in-building automation components including security control for doors/windows, as well as environmental controls and lighting through the use of an in-ceiling, PoE enabled IP/ethernet infrastructure.

EMAN produced version 4 of the "Entity MIB" (ENTITY-MIB) [RFC6933], primarily to introduce globally unique UUIDs for physical entities that allows to better link across different entities, such as a PoE port on an ethernet switch and the device connected to that switch port.

The "Monitoring and Control MIB for Power and Energy" [RFC7460] specifies a MIB for monitoring for Power State and energy consumption of networked. The document discusses the link with other MIBs such as the ENTITY-MIB, the ENTITY-SENSOR-MIB [RFC3433] for which it is amending missing accuracy information to meet IEC power monitoring requirements, the "Power Ethernet MIB" (POWER-ETHERNET-MIB) [RFC3621] to manage PoE, and the pre-existing IETF MIB for Uninterruptable Power Supplies (UPS) (UPS-MIB) [RFC1628], allowing for example to build control systems that manage shutdowns of devices in case of power failure based on UPS battery capacity and device consumptions/priorities. Similarly, the EMAN "Definition of Managed Objects for Battery Monitoring" [RFC7577] defines objects to support battery monitoring in managed devices.

The pre-existing IETF "Entity State MIB" (ENTITY-STATE-MIB) [RFC4268] allows to specify the operational state of entities specified via the ENTITY-MIB respective to their power consumption and operational capabilities (e.g.: "coldStandby", "hotStandby", "ready" etc.). Devices can also act as proxies to provide a MIB interfaces for monitoring and control of power for other devices, that may use other protocols, such as in case of a home gateway interfacing with various vendor specific protocols of home equipment.

The EMAN "Energy Object Context MIB" [RFC7461] defines the ENERGY-OBJECT-CONTEXT-MIB and IANA-ENERGY-RELATION-MIB, both of which serve to "address device identification, context information, and the energy relationships between devices" according to [RFC7461].

To automatically discover and negotiate PoE power consumption between switch and client, non-IETF technologies, such as IEEE "Link Layer Discovery Protocol" (LLDP) and proprietary MIBs for it, such as LLDP-EXT-MED-MIB can be used.
Finally, the "Energy Management (EMAN) Applicability Statement" [RFC7603] provides an overview of EMAN with a user/operator perspective, also reviewing a range of typical scenarios it can support as well as how it could/can link to a variety of pre-existing, non-IETF standards relevant for power management. Such intended applicability includes home, core, and DC networks.

There are currently no YANG equivalent modules. Such modules would not only be designed to echo the EMAN MIBs but would also allow to control dedicated power optimization engines instead of relying upon static and frozen vendor-specific optimization.

8. Power-awareness in Forwarding and Routing Protocols

8.1. Power Aware Networks (PANET)

In 2013/2014, some drafts proposed how networks themselves, specifically those of Internet Service Providers (ISP) could become "power aware" to the extent that its power consumption could be regulated (or self-regulate) based on the current required performance of the network and/or available power, by reducing excess (or too power expensive) network capacity through switching-off/low-powering components such as redundant routers, linecards, interfaces or links, or reducing power consumption by reducing bitrates on links.


The 2014 [I-D.retana-rtgwg-eacp] exemplifies the concept and discusses key challenges such as the reduced resilience against errors when redundant components are switched off, the risk of increased stretch (path length) and therefore latency under partial network component shutdown or downspeeding, as well as the idea of saving energy through (periodic) microsleeps such as possible with "Energy Efficient Ethernet" https://en.wikipedia.org/wiki/Energy-Efficient_Ethernet links. The 2013 draft "Reducing Power Consumption using BGP with power source data", [I-D.mjsraman-panet-inter-as-power-source] proposed BGP attributes to allow calculation of power efficient (or for example green) paths.

One core market driver for this work where rolling blackouts that especially affected India at the time of these drafts, raising the desire to be for example reducing the total power consumption of a network in times of such energy emergencies.
While there was technical interest in the IETF, the market significance for the vendors mostly present in the IETF was considered as not to be important enough. Likewise, traditional routers, unlike for example today's standard PC hardware designs do exhibit little power savings upon shutdown of components such as line-cards or interfaces.

In addition, an SDN / controller-based solution where relatively in their infancy back in 2013/2014, and technologies that would allow for SDN controller to have resilient (self-healing) connectivity such as described in [RFC8368]/[RFC8994] was also not available, making the risk of severely impacting network reliability one of the key factors for this PANET work to not proceed so far.

8.2. SDN-based Semantic Forwarding

Recently, [I-D.boucadair-irtf-sdn-and-semantic-routing] provided the following feature as an examples of capabilities that can be offered by appropriate control of forwarding elements:

Energy-efficient Forwarding: An important effort was made in the past to optimize the energy consumption of network elements. However, such optimization is node-specific and no standard means to optimize the energy consumption at the scale of the network have been defined. For example, many nodes (also, service cards) are deployed as backups.

A controller-based approach can be implemented so that the route selection process optimizes the overall energy consumption of a path. Such a process takes into account the current load, avoids waking nodes/cards for handling "sparse" traffic (i.e., a minor portion of the total traffic), considers node-specific data (e.g., [RFC7460]), etc. This off-line Semantic Routing approach will transition specific cards/nodes to "idle" and wake them as appropriate, etc., without breaking service objectives. Moreover, such an approach will have to maintain an up-to-date topology even if a node is in an "idle" state (such nodes may be removed from adjacency tables if they don't participate in routing advertisements).

8.3. Misc

The non-adopted, expired 2013 draft [I-D.okamoto-ccamp-midori-gmpls-extension-reqs] discusses power awareness in routing in conjunction with Traffic Engineering (tunnels), specifically in the context of Generalized MPLS (GMPLS), e.g.: various L2 technologies such as switched optical fiber networks. It primarily claims the issue that the existing management objects are not sufficient to express energy management related aspects, and
thus do not allow to build energy conscious policies into PCE for such GMPLS networks.

The non-adopted 2013 "Requirements for an Energy-Efficient Network System", [I-D.suzuki-eens-requirements] proposes a signaling of network capacity towards DC, for example based on load or network energy management in support of appropriate performance control (such as VM migration) the DC - or vice versa (DC load-based traffic engineering in the network to support that DC load).

The non-adopted 2013 "Building power optimal Multicast Trees" [I-D.mjsraman-rtgwg-pim-power] proposes that (PIM based) IP Multicast routing could perform local routing choices in the case of "Equal Cost MultiPath" (ECMP) "Reverse Path Forwarding" (RPF) alternatives based on the energy that would be consumed in the router, such as when one ECMP alternative would use a more power efficient linecard or when one ECMP choice was on the same linecard as the interfaces to which the packets would need to be routed (and therefore avoiding to forward the packet across separate ingress and egress linecards).

9. Gaps

The 2013 "Towards an Energy-Efficient Internet" [I-D.winter-energy-efficient-internet] summarizes some of the same work items as this document (as written back in 2013) and lists additional more non-adopted drafts. It also identifies three areas of gaps, that it suggests the IETF to work on: "Load-adaptive Resource Management", "Energy-efficient Protocol Design" and "Energy-efficiency Metrics and Standard Benchmarking Methodologies".

Some aspects for those areas of gaps where partially tackled by later work in the IETF, but broadly speaking, most of those areas remain open to a wide range of possible further IETF/IRTF work.

10. Summary

TBD

11. Changelog

[RFC-Editor: this section to be removed in final document.]

The master for this document is hosted at http://github.com/toerless/energy. Please submit Issues and/or Pull-requests for proposed changes or join the team of authors and edit yourself.

00: Initial version
12. Informative References

[BOUNDED_LATENCY]

[I-D.ajunior-energy-awareness-00]

[I-D.bormann-core-roadmap-05]

[I-D.boucadair-irtf-sdn-and-semantic-routing]

[I-D.castellani-core-alive]
[I-D.chakrabarti-nordmark-energy-aware-nd]

[I-D.claise-power-management-arch]

[I-D.desmouceaux-ipv6-mcast-wifi-power-usage]

[I-D.fossati-core-monitor-option]

[I-D.fossati-core-publish-option]

[I-D.giacomin-core-sleepy-option]
[I-D.ietf-core-coap-pubsub]

[I-D.ietf-dnssd-srp]

[I-D.ietf-tcpm-rfc793bis]

[I-D.jennings-energy-pricing]

[I-D.lhan-problems-requirements-satellite-net]

[I-D.manral-bmwg-power-usage]

[I-D.mjsraman-panet-inter-as-power-source]
Raman, S., Venkataswami, B. V., Raina, G., and K. Veezhinathan, "Reducing Power Consumption using BGP with power source data", Work in Progress, Internet-Draft,
[I-D.mjsraman-rtgwg-pim-power]

[I-D.okamoto-ccamp-midori-gmpls-extension-reqs]

[I-D.petrescu-v6ops-ipv6-power-ipv4]

[I-D.rahman-core-sleepy]

[I-D.rahman-core-sleepy-nodes-do-we-need]

[I-D.rahman-core-sleepy-problem-statement]
[I-D.retana-rtgwg-eacp]

[I-D.suzuki-eens-requirements]

[I-D.vial-core-mirror-proxy]

[I-D.vial-core-mirror-server]

[I-D.wang-roll-energy-optimization-scheme]

[I-D.winter-energy-efficient-internet]

[I-D.zhang-greennet]


Authors’ Addresses

Toerless Eckert (editor)
Futurewei Technologies USA
2220 Central Expressway
Santa Clara, CA 95050
United States of America
Email: tte@cs.fau.de

Mohamed Boucadair
Orange
35000 Rennes
France
Email: mohamed.boucadair@orange.com
Pascal Thubert
Cisco Systems, Inc.
45 Allee des Ormes - BP1200, Building D
06254 MOUGINS Sophia Antipolis
France
Phone: +33 497 23 26 34
Email: pthubert@cisco.com

Jeff Tentsura
Microsoft
Email: jefftant.ietf@gmail.com
A YANG Data Model for OAM and Management of ALTO Protocol
draft-ietf-alto-oam-yang-01

Abstract

This document defines a YANG data model for Operations,
Administration, and Maintenance (OAM) & Management of Application-
Layer Traffic Optimization (ALTO) Protocol. The operator can use the
data model to create and update ALTO information resources, manage
the access control, configure server-to-server communication and
server discovery, and collect statistical data.

Discussion Venues

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

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

Source for this draft and an issue tracker can be found at

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 13 January 2023.

Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.

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

Table of Contents

1. Introduction .............................................. 3
2. Requirements Language .................................... 3
3. Terminology .............................................. 3
   3.1. Tree Diagrams ....................................... 3
   3.2. Prefixes in Data Node Names ....................... 4
4. Design Scope and Requirements ............................. 4
   4.1. Scope of Data Model for ALTO O&M .................... 4
   4.2. Basic Requirements .................................. 5
   4.3. Additional Requirements for Extensibility .......... 6
   4.4. Overview of ALTO O&M Data Model for Reference ALTO
        Architecture ........................................ 6
5. Design of ALTO O&M Data Model ............................. 6
   5.1. Overview of ALTO O&M Data Model .................... 7
   5.2. Meta Information of ALTO Server .................... 8
   5.3. ALTO Information Resources Configuration Management .. 10
   5.4. Data Sources ....................................... 12
      5.4.1. Yang DataStore Data Source ..................... 13
      5.4.2. Prometheus Data Source ........................ 14
   5.5. Model for ALTO Server-to-server Communication .... 14
6. Design of ALTO O&M Statistics Data Model ................. 14
   6.1. Model for ALTO Logging and Fault Management ....... 14
   6.2. Model for ALTO-specific Performance Monitoring .... 14
7. Extension of ALTO O&M Data Model ......................... 16
8. ALTO OAM YANG Module ..................................... 17
   8.1. The ietf-alto Module ................................ 17
   8.2. The ietf-alto-stats Module ......................... 34
9. Security Considerations .................................. 39
10. IANA Considerations ...................................... 39
11. References .............................................. 39
   11.1. Normative References .............................. 40
This document defines a YANG data model for the Operations, Administration, and Maintenance (OAM) & Management of Application-Layer Traffic Optimization (ALTO) Protocol. The basic purpose of this YANG data model is discussed in Section 16 of [RFC7285]. The operator can use the data model to create and update ALTO information resources, manage the access control, configure server-to-server communication and server discovery, and collect statistical data.

The basic structure of this YANG data model is guided by Section 16 of [RFC7285] and [RFC7971]. Although the scope of the YANG data model in this document mainly focuses on the support of the base ALTO protocol [RFC7285] and the existing ALTO standard extensions (including [RFC8189], [RFC8895] and [RFC8896]), the design will also be extensible for future standard extensions (e.g., [I-D.ietf-alto-path-vector], [I-D.ietf-alto-unified-props-new], [I-D.ietf-alto-cdni-request-routing-alto], and [I-D.ietf-alto-performance-metrics]).

2. Requirements Language

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

3. Terminology

This document uses the following acronyms:

* OAM - Operations, Administration, and Maintainance
* O&M - OAM and Management

3.1. Tree Diagrams

A simplified graphical representation of the data model is used in this document. The meaning of the symbols in these diagrams is defined in [RFC8340].
3.2. Prefixes in Data Node Names

In this document, names of data nodes and other data model objects are often used without a prefix, as long as it is clear from the context in which YANG module each name is defined. Otherwise, names are prefixed using the standard prefix associated with the corresponding YANG module, as shown in Table 1.

<table>
<thead>
<tr>
<th>Prefix</th>
<th>YANG module</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>yang</td>
<td>ietf-yang-types</td>
<td>[RFC6991]</td>
</tr>
<tr>
<td>inet</td>
<td>ietf-inet-types</td>
<td>[RFC6991]</td>
</tr>
<tr>
<td>key-chain</td>
<td>ietf-key-chain</td>
<td>[RFC8177]</td>
</tr>
</tbody>
</table>

Table 1: Prefixes and corresponding YANG modules

4. Design Scope and Requirements

4.1. Scope of Data Model for ALTO O&M

What is in the scope of this document?

* Data model for deploy an ALTO server/client.
* Data model for operate and manage a running ALTO server/client.
* Data model for functionality/capability configuration for ALTO services.
* Data model for monitoring ALTO-related performance metrics.

What is not in the scope of this document?

This document does not define any data model related to specific implementation, including:

* Data structures for how to store/deliver ALTO information resources (e.g., database schema to store a network map).
* Data structures for how to store information collected from data sources. (e.g., database schema to store topology collected from an Interface to the Routing System (I2RS) client [RFC7921])
4.2. Basic Requirements

Based on discussions and recommendations in [RFC7285] and [RFC7971], the data model provided by this document satisfies basic requirements listed in Table 2.

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1: The data model should support configuration for ALTO server setup.</td>
<td>Section 16.1 of [RFC7285]</td>
</tr>
<tr>
<td>R2: The data model should provide logging management.</td>
<td>Section 16.2.1 of [RFC7285]</td>
</tr>
<tr>
<td>R3: The data model should provide ALTO-related management information.</td>
<td>Section 16.2.2 of [RFC7285]</td>
</tr>
<tr>
<td>R4: The data model should provide metrics for server failures.</td>
<td>Section 16.2.3 of [RFC7285], Section 3.3 of [RFC7971]</td>
</tr>
<tr>
<td>R5-1: The data model should support configuration for different data sources.</td>
<td>Section 16.2.4 of [RFC7285], Section 3.2 of [RFC7971]</td>
</tr>
<tr>
<td>R5-2: The data model should support configuration for information resource generation algorithms.</td>
<td>Section 16.2.4 of [RFC7285]</td>
</tr>
<tr>
<td>R5-3: The data model should support configuration for access control at information resource level.</td>
<td>Section 16.2.4 of [RFC7285]</td>
</tr>
<tr>
<td>R6: The data model should provide performance monitoring for ALTO-specific metrics.</td>
<td>Section 16.2.5 of [RFC7285], Section 3.4 of [RFC7971]</td>
</tr>
<tr>
<td>R7: The data model should support configuration for security policy management.</td>
<td>Section 16.2.6 of [RFC7285]</td>
</tr>
</tbody>
</table>

Table 2: Basic Requirements of Data Model for ALTO O&M.
4.3. Additional Requirements for Extensibility

R8: As the ALTO protocol is extensible, the data model for ALTO O&M should allow for augmentation to support potential future extensions.

4.4. Overview of ALTO O&M Data Model for Reference ALTO Architecture

Figure 1 shows a reference architecture for ALTO server implementation and YANG modules that server components implement. The server manager, information resource manager and data source listeners need to implement ietf-alto.yang (see Section 5). The performance monitor and logging and fault manager need to implement ietf-alto-stats.yang (see Section 6).

The data broker and algorithm plugins are not in the scope of the data model defined in this document. But user specified YANG modules can be applied to different algorithm plugins by augmenting the data model defined in this document (see Section 7).

Figure 1: A Reference ALTO Server Architecture and YANG Modules

5. Design of ALTO O&M Data Model
5.1. Overview of ALTO O&M Data Model

The ietf-alto module defined in this document provide all the basic ALTO O&M data models fitting the requirements listed in Section 4.

The container "alto-server" in the ietf-alto module contains all the configured and operational parameters of the administered ALTO server instance.

NOTE: So far, the ALTO YANG module only focuses on the ALTO server related configuration. The ALTO client related configuration will be added in a future version of the document.

module: ietf-alto
  +--rw alto-server
    |  |     +--rw listen
    |  |    |  |  +--rw alto-server-listen-stack-grouping
    |  |      +--rw cost-type* [cost-type-name]
    |  |       |  +--rw cost-type-name string
    |  |       +--rw cost-mode identityref
    |  |       +--rw cost-metric identityref
    |  |      +--rw meta* [meta-key]
    |  |          |  +--rw meta-key string
    |  |          +--rw meta-value string
    |  +--rw resource* [resource-id]
    |      +--rw resource-id resource-id
    |      +--rw resource-type identityref
    |      +--rw description? string
    |      +--rw accepted-group* string
    |      +--rw dependency* resource-id
    |      +--rw auth
    |          |  |  +--(auth-type-selection)?
    |          |      |  +--:(auth-key-chain)
    |          |      |      +--rw key-chain? key-chain:key-chain-ref
    |          |      +--:(auth-key)
    |          |      +--:(auth-tls)
    |          +--(resource-params)?
    |              +--:(ird)
    |                  +--rw delegation inet:uri
    |                      +--:(networkmap)
    |                          +--rw alto-networkmap-params
    |                              +--rw is-default? boolean
    |                              +--rw filtered? boolean
    |                              +--u algorithm
    |                          +--:(costmap)
    |                              +--rw alto-costmap-params
    |                                  +--rw filtered? boolean
5.2. Meta Information of ALTO Server

The ALTO server instance contains the following basic configurations for the server setup.
The "listen" contains all the configurations for the whole server listen stack across HTTP layer, TLS layer and TCP layer.

```yaml
grouping alto-server-listen-stack-grouping
  +-- (transport)
    +--:(http) {http-listen}?
      +-- http
        |  +-- tcp-server-parameters
        |      +---u tcp:tcp-server-grouping
        |      +-- http-server-parameters
        |      +-- alto-server-parameters
        +--:(https)
          +-- https
            +-- tcp-server-parameters
            +---u tcp:tcp-server-grouping
            +-- tls-server-parameters
            |  +---u tls:tls-server-grouping
            +-- http-server-parameters
            |  +---u http:http-server-grouping
            +-- alto-server-parameters

TODO: A "base-uri" for ALTO clients to access may still be needed.

The "cost-type" list is the registry for the cost types that can be used in the ALTO server.

The "meta" list contains the customized meta data of the ALTO server. It will be populated into the meta field of the default Information Resource Directory (IRD).

TODO: As suggested by [RFC7286] and [RFC8686], the configuration related to ALTO server discovery should also be included here.

module: ietf-alto
  +--rw alto-server
    +--rw listen
      |  +---u alto-server-listen-stack-grouping
    +--rw cost-type* [cost-type-name]
      |  +--rw cost-type-name  string
      |  +--rw cost-mode  identityref
      |  +--rw cost-metric  identityref
    +--rw meta* [meta-key]
      |  +--rw meta-key  string
      |  +--rw meta-value  string
```
5.3. ALTO Information Resources Configuration Management

The ALTO server instance contains a list of resource entries. Each resource entry contains the configurations of an ALTO information resource (See Section 8.1 of [RFC7285]). The operator of the ALTO server can use this model to create, update, and remove the ALTO information resource.

Each resource entry provide configuration defining how to create or update an ALTO information resource. Adding a new resource entry will submit an ALTO information resource creation intent to the intent system to create a new ALTO information resource. Updating an existing resource entry will update the corresponding ALTO information resource creation intent. Removing an existing resource entry will remove the corresponding ALTO information resource creation intent and also the created ALTO information resource.

The parameter of the intent interface defined by a resource entry MUST include a unique resource-id and a resource-type.

It can also include an accepted-group node containing a list of user-groups that can access this ALTO information resource.

As section 15.5.2 of [RFC7285] suggests, the module also defines authentication related configuration to employ access control at information resource level. The ALTO server returns the IRD to the ALTO client based on its authentication information.

For some resource-type, the parameter of the intent interface MUST also include the a dependency node containing the resource-id of the dependent ALTO information resources (See Section 9.1.5 of [RFC7285]).

For each type of ALTO information resource, the creation intent MAY also need type-specific parameters. These type-specific parameters include two categories:

1. One categories of the type-specific parameters are common for the same type of ALTO information resource. They declare the Capabilities of the ALTO information resource (See Section 9.1.3 of [RFC7285]).

2. The other categories of the type-specific parameters are algorithm-specific. The developer of the ALTO server can implement their own creation algorithms and augment the algorithm node to declare algorithm-specific input parameters.
Except for the ird resource, all the other types of resource entries have augmented algorithm node. The augmented algorithm node can reference data sources subscribed by the data-source entries (See Section 5.4).

The developer cannot customize the creation algorithm of the ird resource. The default ird resource will be created automatically based on all the added resource entries. The delegated ird resource will be created as a static ALTO information resource (See Section 9.2.4 of [RFC7285]).

module: ietf-alto
  +++-rw alto-server
  ...
  +++-rw resource* [resource-id]
      +++-rw resource-id resource-id
      +++-rw resource-type identityref
      +++-rw description? string
      +++-rw accepted-group* string
      +++-rw dependency* resource-id
      +++-rw auth
          +++-rw (auth-type-selection)?
              +++-:(auth-key-chain)
                  +++-rw key-chain? key-chain:key-chain-ref
              +++-:(auth-key)
              +++-:(auth-tls)
      +++-rw (resource-params)?
          +++-:(ird)
              +++-rw alto-ird-params
              +++-rw delegation inet:uri
          +++-:(networkmap)
              +++-rw alto-networkmap-params
              +++-rw is-default? boolean
              +++-rw filtered? boolean
              +++-u algorithm
          +++-:(costmap)
              +++-rw alto-costmap-params
              +++-rw filtered? boolean
              +++-u filt-costmap-cap
              +++-u algorithm
          +++-:(endpointcost)
              +++-rw alto-endpointcost-params
              +++-u filt-costmap-cap
              +++-u algorithm
          +++-:(endpointprop)
              +++-rw alto-endpointprop-params
              +++-rw prop-types* string
              +++-u algorithm
5.4. Data Sources

The ALTO server instance contains a list of data-source entries to subscribe the data sources from which ALTO information resources are derived (See Section 16.2.4 of [RFC7285]).

A data-source entry MUST include:

* a unique source-id for resource creation algorithms to reference,
* the source-type attribute to declare the type of the data source,
* the update-policy to specify how to get the data update from the data source,
* the source-params to specify where and how to query the data.

The update policy can be either reactive or proactive. For the reactive update, the ALTO server gets the update as soon as the data source changes. For the proactive update, the ALTO server has to proactively fetch the data source periodically.

To use the reactive update, the reactive attribute MUST be set true. To use the proactive update, the poll-interval attribute MUST be greater than zero. The value of poll-interval specifies the interval of fetching the data in milliseconds. If reactive is false or poll-interval is zero, the ALTO server will not update the data source.
The data-source/source-params node can be augmented for different types of data sources. This data model only includes import interfaces for a list of predefined data sources. More data sources can be supported by future documents and other third-party providers.

module: ietf-alto
   +--rw alto-server
      ...
      +--rw data-source* [source-id]
         +--rw source-id            string
         +--rw source-type          identityref
         +--rw (update-policy)
            |   +--rw reactive         boolean
            +--:(proactive)
               +--rw poll-interval     uint32
         +--rw (source-params)?
            +--:(yang-datastore)
               +--rw yang-datastore-source-params
               |   +--rw source-path
               |      yang:xpath1.0
               +--rw (restconf-endpoint)?
                  +--:(local)
                  +--:(remote)
                     +--rw restconf-endpoint-params
                        +---u rcc:restconf-client-listen-stack-grouping
            +--:(prometheus)
               +--rw prometheus-source-params
                  +--rw source-uri    inet:uri
                  +--rw query-data?   string

Note: Current source configuration still has limitations. It should be revised to support more general southbound and data retrieval mechanisms.

5.4.1. Yang DataStore Data Source

The yang-datastore-source-params is used to import the YANG data from a YANG model-driven data store.

It supports two types of endpoints: local and remote.

* For a local endpoint, the YANG data is located the data from the same YANG model-driven data store supplying the current ALTO O&M data model. Therefore, the ALTO data source listener retrieves the data using the internal API provided by the data store.
* For a remote endpoint, the ALTO data source listener establishes an HTTP connection to the remote RESTCONF server, and retrieve the data using the RESTCONF API.

The source-path is used to specify the XPath of the data source node.

5.4.2. Prometheus Data Source

The prometheus-source-params is used to import common performance metrics data which is provided by a Prometheus server. The source-url is used to establish the connection with the Prometheus server. The query-data is used to specify the potential query expression in PromQL.

5.5. Model for ALTO Server-to-server Communication

In practice, multiple ALTO servers can be deployed for scalability. That may require communication among different ALTO servers.

The YANG module defined in this document contains the configuration for the communication between two ALTO servers.

TODO: this is still under the open discussion status.

6. Design of ALTO O&M Statistics Data Model

6.1. Model for ALTO Logging and Fault Management

As section 16.2.1 and section 16.2.3 of [RFC7285] suggest, the YANG data module defined in this document contains statistics for logging and failure detection.

NOTE: The detailed YANG module will appear in the future version.

6.2. Model for ALTO-specific Performance Monitoring

As section 16.2.5 of [RFC7285] suggests, the YANG data module defined in this document also contains statistics for ALTO-specific performance metrics.

More specifically, this data model contains the following measurement information suggested by [RFC7971]:

* Measurement of impact
  - Total amount and distribution of traffic
  - Application performance
* System and service performance
  - Requests and responses for each information resource
  - CPU and memory utilization
  - ALTO map updates
  - Number of PIDs
  - ALTO map sizes

Besides the measurement information suggested by [RFC7971], this data model also contains useful measurement information for other ALTO extensions:

* Number of generic ALTO entities (for
  [I-D.ietf-alto-unified-props-new] and
  [I-D.ietf-alto-cdni-request-routing-alto])

* Statistics for update sessions and events (for [RFC8189])

* Statistics for calendar (for [RFC8896])

The module, "ietf-alto-stats", augments the ietf-alto module to include statistics at the ALTO server and information resource level.
module: ietf-alto-stats

augment /alto:alto-server:
  +--ro num-total-req?         yang:counter32
  +--ro num-total-succ?        yang:counter32
  +--ro num-total-fail?        yang:counter32
  +--ro num-total-last-req?    yang:counter32
  +--ro num-total-last-succ?   yang:counter32
  +--ro num-total-last-fail?   yang:counter32

augment /alto:alto-server/alto:resource:
  +--ro num-res-upd?    yang:counter32
  +--ro res-mem-size?   yang:counter32
  +--ro res-enc-size?   yang:counter32
  +--ro num-res-req?    yang:counter32
  +--ro num-res-succ?   yang:counter32
  +--ro num-res-fail?   yang:counter32

  /alto:networkmap/alto:alto-networkmap-params:
  +--ro num-map-pid?   yang:counter32

  /alto:propmap/alto:alto-propmap-params:
  +--ro num-map-entry?   yang:counter32

  /alto:cdni/alto:alto-cdni-params:
  +--ro num-base-obj?   yang:counter32

  /alto:update/alto:alto-update-params:
  +--ro num-upd-sess?      yang:counter32
  +--ro num-event-total?   yang:counter32
  +--ro num-event-max?     yang:counter32
  +--ro num-event-min?     yang:counter32
  +--ro num-event-avg?     yang:counter32

7. Extension of ALTO O&M Data Model

As ALTO protocol is extensible, new protocol extensions can be
developed after this data model is published. To support future ALTO
protocol extensions, the extension documents can augment the existing
cases of the resource-params choice with new configuration parameters
for existing ALTO information resource extensions, or augment the
resource-params with new cases for new ALTO information resources.

Developers and operators can also extend this ALTO O&M data model to
align with their own implementations. Specifically, the following
nodes of the data model can be augmented:

* The algorithm choice of the resource-params of each resource.
* The data-source choice.

The following example shows how the developer augments the algorithm choice of alto-networkmap-params with a creation algorithm for the network map resource.

module: example-ietf-alto-alg

  /alto:networkmap/alto:alto-networkmap-params
  /alto:algorithm:
    +--:(l3-unicast-cluster)
    +--rw l3-unicast-cluster-algorithm
    +--rw l3-unicast-topo
      |   -> /alto:alto-server/data-source/source-id
    +--rw depth?             uint32

This example defines a creation algorithm called l3-unicast-cluster-algorithm for the network map resource. It takes two algorithm-specific parameters:

l3-unicast-topo  This parameter refers to the source id of a data source node subscribed in the data-source list (See Section 5.4). The corresponding data source is assumed to be an internet data source (See Section 5.4.1) for an IETF layer 3 unicast topology defined in [RFC8346]. The algorithm uses the topology data from this data source to compute the ALTO network map resource.

depth  This optional parameter sets the depth of the clustering algorithm. For example, if the depth sets to 1, the algorithm will generate PID for every l3-node in the topology.

The creation algorithm can be reactively called once the referenced data source updates. Therefore, the ALTO network map resource can be updated dynamically. The update of the reference data source depends on the used update-policy (See Section 5.4).

8. ALTO OAM YANG Module

8.1. The ietf-alto Module

<CODE BEGINS> file "ietf-alto@2022-07-11.yang"
module ietf-alto {
  yang-version 1.1;
  namespace
    "urn:ietf:params:xml:ns:yang:ietf-alto";
  prefix "alto";
import ietf-inet-types {
  prefix "inet";
  reference
    "RFC 6991: Common YANG Data Types";
}

import ietf-yang-types {
  prefix "yang";
  reference
    "RFC 6991: Common YANG Data Types";
}

import ietf-key-chain {
  prefix key-chain;
  reference
    "RFC 8177: YANG Data Model for Key Chains";
}

import ietf-tcp-server {
  prefix tcp;
  reference
    "RFC DDDD: YANG Groupings for TCP Clients and TCP Servers";
}

import ietf-tls-server {
  prefix tls;
  reference
    "RFC FFFF: YANG Groupings for TLS Clients and TLS Servers";
}

import ietf-http-server {
  prefix http;
  reference
    "RFC GGGG: YANG Groupings for HTTP Clients and HTTP Servers";
}

import ietf-restconf-client {
  prefix rcc;
  reference
    "RFC IIII: YANG Groupings for RESTCONF Clients and RESTCONF
     Servers";
}

organization
  "IETF ALTO Working Group";

contact
  "WG Web:  <https://datatracker.ietf.org/wg/alto/about/>"
This YANG module defines all the configured and operational parameters of the administrated ALTO server instance.

Copyright (c) 2022 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 (https://www.rfc-editor.org/info/rfcXXXX); see the RFC itself for full legal notices.

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 (RFC 2119) (RFC 8174) when, and only when, they appear in all capitals, as shown here.

revision "2022-07-11" {
  description
    "Initial Version.";
  reference
    "RFC XXXX: A YANG Data Model for OAM and Management of ALTO Protocol.";
}

typedef resource-id {
  type string {
    length "1..64";
    pattern "[0-9a-zA-Z\-:@\-_]*";
  }
  description
    "Format of Resource ID";
  reference
    "Section 9.1.1 of RFC 7285.";
}

// Base identities

identity resource-type {

description
    "Base identity for type of information resource."
}

identity source-type {
    description
        "Base identity for type of data source."
}

identity cost-mode {
    description
        "The cost mode attribute indicates how costs should be interpreted. Specifically, the cost mode attribute indicates whether returned costs should be interpreted as numerical values or ordinal rankings."
    reference
        "Section 6.1.2 of RFC 7285."
}

identity cost-metric {
    description
        "The cost metric attribute indicates what the cost represents."
    reference
        "Section 6.1.1 of RFC 7285."
}

// Identities for ALTO information resources

identity network-map {
    base resource-type;
    description
        "Identity for network map."
}

identity cost-map {
    base resource-type;
    description
        "Identity for cost map."
}

identity property-map {
    base resource-type;
    description
        "Identity for property map."
}

// Identities for predefined data sources
identity yang-datastore {
    base source-type;
    description
        "Identity for data source of YANG-based datastore.";
}

identity prometheus {
    base source-type;
    description
        "Identity for data source of prometheus system.";
}

// Identities for cost mode

identity numerical {
    base cost-mode;
    description
        "This mode indicates that it is safe to perform numerical operations";
}

identity ordinal {
    base cost-mode;
    description
        "This mode indicates that the cost values in a cost map represent ranking";
}

identity array {
    if-feature "path-vector";
    base cost-mode;
    description
        "This mode indicates that every cost value in the response body of a (Filtered) Cost Map or an Endpoint Cost Service MUST be interpreted as a JSON array.";
}

// Identities for cost metrics

identity routingcost {
    base cost-metric;
    description
        "This metric conveys a generic measure for the cost of routing traffic from a source to a destination.";
}

identity ane-path {
    if-feature "path-vector";
}
This metric indicates that the value of such a cost type conveys an array of Abstract Network Element (ANE) names, where each ANE name uniquely represents an ANE traversed by traffic from a source to a destination.

identity delay-ow {
  if-feature "performance-metrics";
  base cost-metric;
  description
    "Section 4.1 of RFC XXXX";
}

identity delay-rt {
  if-feature "performance-metrics";
  base cost-metric;
  description
    "Section 4.2 of RFC XXXX";
}

identity delay-variation {
  if-feature "performance-metrics";
  base cost-metric;
  description
    "Section 4.3 of RFC XXXX";
}

identity lossrate {
  if-feature "performance-metrics";
  base cost-metric;
  description
    "Section 4.4 of RFC XXXX";
}

identity hopcount {
  if-feature "performance-metrics";
  base cost-metric;
  description
    "Section 4.5 of RFC XXXX";
}

identity tput {
  if-feature "performance-metrics";
  base cost-metric;
  description
    "Section 5.1 of RFC XXXX";
identity bw-residual {
    if-feature "performance-metrics";
    base cost-metric;
    description
        "Section 5.2 of RFC XXXX";
}

identity bw-available {
    if-feature "performance-metrics";
    base cost-metric;
    description
        "Section 5.3 of RFC XXXX";
}

// Features

feature http-listen {
    description
        "The 'http-listen' feature is only used for test deployment. According to Sec 8.3.5 of RFC 7285, it shouldn’t be used in the production deployment.";
}

feature multi-cost {
    description
        "Support multi-cost extension.";
    reference
        "RFC 8189: Multi-Cost Application-Layer Traffic Optimization (ALTO)";
}

feature incr-update {
    description
        "Support incremental update extension.";
    reference
        "RFC 8895: Application-Layer Traffic Optimization (ALTO) Incremental Updates Using Server-Sent Events (SSE)";
}

feature cost-calendar {
    description
        "Support cost calendar extension.";
    reference
        "RFC 8896: Application-Layer Traffic Optimization (ALTO) Cost Calendar";
}
feature propmap {
    description
    "Support entity property map extension.";
    reference
    "RFC 9240: An ALTO Extension: Entity Property Maps";
}

feature cdni {
    description
    "Support CDNi extension.";
    reference
    "RFC 9241: Content Delivery Network Interconnection (CDNi) Request Routing: CDNI Footprint and Capabilities Advertisement using ALTO";
}

feature path-vector {
    description
    "Support path vector extension.";
    reference
    "RFC XXXX: An ALTO Extension: Path Vector";
}

feature performance-metrics {
    description
    "Support performance metrics extension.";
    reference
    "RFC XXXX: ALTO Performance Cost Metrics";
}

// Groupings

grouping filt-costmap-cap {
    description
    "This grouping defines data model for FilteredCostMapCapabilities.";
    reference
    "Sec 11.3.2.4 of RFC 7285.";
    leaf-list cost-type-names {
        type string;
        min-elements 1;
        description
        "Supported cost types";
    }
    leaf cost-constraints {
        type boolean;
        description
        "If true, then the ALTO server allows cost
constraints to be included in requests to the corresponding URI. If not present, this field MUST be interpreted as if it specified false.

leaf max-cost-types {
  if-feature "multi-cost";
  type uint32;
  default 0;
  description
    "If present with value N greater than 0, this resource understands the multi-cost extensions in this document and can return a multi-cost map with any combination of N or fewer cost types in the 'cost-type-names' list. If omitted, the default value is 0."
}

leaf-list testable-cost-type-names {
  if-feature "multi-cost";
  type string;
  description
    "If present, the resource allows constraint tests, but only on the cost type names in this array."
}

container calendar-attributes {
  if-feature "cost-calendar";
  leaf-list cost-type-names {
    type string;
    min-elements 1;
    description
      "An array of one or more elements indicating the cost type names in the IRD entry to which the values of 'time-interval-size' and 'number-of-intervals' apply."
  }
  leaf time-interval-size {
    type decimal64 {
      fraction-digits 4;
    }
    mandatory true;
    description
      "The duration of an ALTO Calendar time interval in a unit of seconds."
  }
  leaf number-of-intervals {
    type uint32 {
      range "1..max";
    }
    mandatory true;
    description
      "A strictly positive integer (greater or equal to 1) that
indicates the number of values of the Cost Calendar array.


"Configuration for CalendarAttributes.";
reference
"Section 4.1 of RFC 8896."
}

}

}

}

}

}

}

}

}

choice algorithm {
    mandatory true;
    description
    "Information resource creation algorithm to be augmented.";
}

description
"This grouping defines base data model for information resource creation algorithm."

}

}

}

}

}

choice transport {
    mandatory true;
    description
    "Selects between available transports.";
    case http {
        if-feature "http-listen";
        container http {
            description
            "Configures ALTO server stack assuming that TLS-termination is handled externally.";
            container tcp-server-parameters {
                description
                "A wrapper around the TCP server parameters to avoid name collisions.";
                uses tcp:tcp-server-grouping {

refine "local-port" {
  default "80";
  description
  "The RESTCONF server will listen on the IANA-assigned well-known port value for 'http' (80) if no value is specified.";
}
}
}

container http-server-parameters {
  description
  "A wrapper around the HTTP server parameters to avoid name collisions.";
  uses http:http-server-grouping;
}

container alto-server-parameters {
  description
  "A wrapper around the ALTO server parameters to avoid name collisions.";
  uses alto-server-grouping;
}

}

case https {
  container https {
    description
    "Configures ALTO server stack assuming that TLS-termination is handled internally.";
  container tcp-server-parameters {
    description
    "A wrapper around the TCP server parameters to avoid name collisions.";
    uses tcp:tcp-server-grouping {
      refine "local-port" {
        default "443";
        description
        "The ALTO server will listen on the IANA-assigned well-known port value for 'https' (443) if no value is specified.";
      }
    }
    container tls-server-parameters {
      description
      "A wrapper around the TLS server parameters to avoid name collisions.";
      uses tls:tls-server-grouping;
    }
    container http-server-parameters {
    description
    "A wrapper around the HTTP server parameters to avoid name collisions.";
    uses http:http-server-grouping;
    }
  }
}
}
// Top-level container

container alto-server {
    description
        "The ALTO server instance.";
    container listen {
        description
            "Configure the ALTO server to listen for ALTO clients.";
        uses alto-server-listen-stack-grouping;
    }
    list cost-type {
        key "cost-type-name";
        leaf cost-type-name {
            type string;
            description
                "The name to reference cost type";
        }
        leaf cost-mode {
            type identityref {
                base cost-mode;
            }
            mandatory true;
            description
                "The referenced cost mode";
        }
        leaf cost-metric {
            type identityref {
                base cost-metric;
            }
            mandatory true;
            description
                "The referenced cost metric";
        }
    }
}
"The referenced cost metric";
}
description
 "Mapping between name and referenced cost type";
}
list meta {
  key "meta-key";
  leaf meta-key {
    type string;
    description
     "Custom meta key";
  }
  leaf meta-value {
    type string;
    mandatory true;
    description
     "Custom meta value";
  }
}
description
 "Mapping of custom meta information";
reference
 "Section 8.4.1 of RFC 7285.";
}
list resource {
  key "resource-id";
  leaf resource-id {
    type resource-id;
    description
     "resource-id to be defined.";
  }
  leaf resource-type {
    type identityref {
      base resource-type;
      mandatory true;
      description
       "identityref to be defined.";
    }
  }
  leaf description {
    type string;
    description
     "The optional description for this information resource.";
  }
  leaf-list accepted-group {
    type string;
    description
     "Access list for authenticated clients.";
  }
}
leaf-list dependency {
  type resource-id;
  description
    "A list of dependent information resources.";
}

container auth {
  description
    "The authentication options";
  choice auth-type-selection {
    description
        "Options for expressing authentication setting.";
    case auth-key-chain {
      leaf key-chain {
        type key-chain:key-chain-ref;
        description
          "key-chain name.";
      }
    }
    case auth-key {
    }
    case auth-tls {
    }
  }
}

choice resource-params {
  description
    "Resource-specific configuration.";
  case ird {
    container alto-ird-params {
      leaf delegation {
        type inet:uri;
        mandatory true;
        description
          "Upstream IRD to be delegated";
      }
      description
        "IRD-specific configuration";
    }
  }
  case networkmap {
    container alto-networkmap-params {
      description
        "(Filtered) Network Map specific configuration";
      reference
        "Section 11.2.1 and Section 11.3.1 of RFC 7285.";
      leaf is-default {
        type boolean;
description
    "Set whether this is the default network map";
}
leaf filtered {
    type boolean;
    default false;
    description
    "Configure whether filtered network map is supported.";
}
uses algorithm;
}
}
case costmap {
    container alto-costmap-params {
        description
        "(Filtered) Cost Map specific configuration";
        reference
        "Section 11.2.2 and Section 11.3.2 of RFC 7285.";
        leaf filtered {
            type boolean;
            description
            "Configure whether filtered cost map is supported.";
        }
        uses filt-costmap-cap;
        uses algorithm;
    }
}
case endpointcost {
    container alto-endpointcost-params {
        description
        "Endpoint Cost Service specific configuration";
        reference
        "Section 11.5 of RFC 7285";
        uses filt-costmap-cap;
        uses algorithm;
    }
}
case endpointprop {
    container alto-endpointprop-params {
        description
        "Endpoint Cost Service specific configuration";
        reference
        "Section 11.5 of RFC 7285";
        leaf-list prop-types {
            type string;
            min-elements 1;
            description
        }
    }
}
"Supported endpoint properties."
}
  uses algorithm;
}
}
case propmap {
  if-feature "propmap"
  container alto-propmap-params {
    uses algorithm;
    description
      "(Filtered) Entity Property Map specific configuration";
  }
}
case cdni {
  if-feature "cdni"
  container alto-cdni-params {
    uses algorithm;
    description
      "CDNi specific configuration";
  }
}
case update {
  if-feature "incr-update"
  container alto-update-params {
    uses algorithm;
    description
      "Incremental Updates specific configuration";
  }
}
}
description
  "ALTO information resources to be defined";
}
list data-source {
  key "source-id";
  leaf source-id {
    type string;
    description
      "Data source id that can be referenced by information resource creation algorithms.";
  }
  leaf source-type {
    type identityref {
      base source-type;
    }
    mandatory true;
    description

"Source-type to be defined";
}
choice update-policy {
    mandatory true;
    case reactive {
        leaf reactive {
            type boolean;
            mandatory true;
            description "Reactive mode";
        }
    }
    case proactive {
        leaf poll-interval {
            type uint32;
            mandatory true;
            description "Polling interval in seconds for proactive mode";
        }
    }
}
description "Policy to get updates from data sources";
choice source-params {
    case yang-datastore {
        container yang-datastore-source-params {
            leaf source-path {
                type yang:xpath1.0;
                mandatory true;
                description "XPath to subscribed YANG datastore node";
            }
        }
        description "YANG datastore specific configuration";
        choice restconf-endpoint {
            case local {
                // Use local API to access YANG datastore
            }
            case remote {
                container restconf-endpoint-params {
                    uses rcc:restconf-client-listen-stack-grouping;
                }
            }
        }
    }
    case prometheus {
        container prometheus-source-params {

leaf source-uri {
    type inet:uri;
    mandatory true;
    description "URI to prometheus agent";
}
leaf query-data {
    type string;
    description "Query expression";
}

8.2. The ietf-alto-stats Module

<CODE BEGINS> file "ietf-alto-stats@2022-07-11.yang"
module ietf-alto-stats {
    yang-version 1.1;
    prefix "alto-stats";

    import ietf-yang-types {
        prefix "yang";
        reference "RFC 6991: Common YANG Data Types";
    }

    import ietf-alto {
        prefix alto;
        reference "RFC XXXX: A YANG Data Model for OAM and Management of ALTO Protocol.";
    }

    organization
This YANG module defines all the configured and operational parameters of the administrated ALTO server instance.

Copyright (c) 2022 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 (https://www.rfc-editor.org/info/rfcXXXX); see the RFC itself for full legal notices.

revision "2022-07-11" {
  description
    "Initial Version.";
  reference
    "RFC XXXX: A YANG Data Model for Operations, Administration, and Maintenance of ALTO Protocol.";
}

augment "/alto:alto-server" {
  description
    "Top-level statistics for the whole ALTO server.";
  leaf num-total-req {
    type yang:counter32;
    config false;
    description
      "The total number of ALTO requests received by this ALTO server.";
  }
  leaf num-total-succ {
    type yang:counter32;
    config false;
    description
      "The total number of successful responses sent by this ALTO server.";
  }
}
leaf num-total-fail {
    type yang:counter32;
    config false;
    description
    "The total number of failed responses sent by this ALTO server.";
}
leaf num-total-last-req {
    type yang:counter32;
    config false;
    description
    "The total number of ALTO requests received within the last 5 minutes.";
}
leaf num-total-last-succ {
    type yang:counter32;
    config false;
    description
    "The total number of successful responses sent by this ALTO server within the last 5 minutes.";
}
leaf num-total-last-fail {
    type yang:counter32;
    config false;
    description
    "The total number of failed responses sent by this ALTO server within the last 5 minutes.";
}

augment "/alto:alto-server/alto:resource" {
    description
    "Common statistics for each information resource.";
    leaf num-res-upd {
        type yang:counter32;
        config false;
        description
        "The number of version updates since the information resource was created.";
    }
    leaf res-mem-size {
        type yang:counter32;
        config false;
        description
        "Memory size (Bytes) utilized by the information resource.";
    }
    leaf res-enc-size {

type yang:counter32;
config false;
description
"Size (Bytes) of JSON encoded data of the information resource.";
}
leaf num-res-req {
  type yang:counter32;
  config false;
description
  "The number of ALTO requests to this information resource.";
}
leaf num-res-succ {
  type yang:counter32;
  config false;
description
  "The number of successful responses for requests to this information resource.";
}
leaf num-res-fail {
  type yang:counter32;
  config false;
description
  "The total number of failed responses for requests to this information resource.";
}
}

augment "/alto:alto-server/alto:resource/alto:resource-params"
  + "/alto:networkmap/alto:alto-networkmap-params" {
  description
  "Augmented statistics for network maps only.";
  leaf num-map-pid {
    type yang:counter32;
    config false;
description
    "Number of PIDs contained by the network map.";
  }
}

augment "/alto:alto-server/alto:resource/alto:resource-params"
  + "/alto:propmap/alto:alto-propmap-params" {
  description
  "Augmented statistics for property maps only.";
  leaf num-map-entry {
    type yang:counter32;
    config false;
description
  }
}
"Number of ALTO entities contained by the property map.";
}
}
augment "/alto:alto-server/alto:resource/alto:resource-params"
  + "/alto:cdni/alto:alto-cdni-params" {
  description
  "Augmented statistics for CDNi resources only.";
  leaf num-base-obj {
    type yang:counter32;
    config false;
    description
    "Number of base CDNi advertisement objects contained by the
    CDNi resource.";
  }
}

augment "/alto:alto-server/alto:resource/alto:resource-params"
  + "/alto:update/alto:alto-update-params" {
  description
  "Augmented statistics for incremental updates only.";
  leaf num-upd-sess {
    type yang:counter32;
    config false;
    description
    "Number of sessions connected to the incremental update
    service.";
  }
  leaf num-event-total {
    type yang:counter32;
    config false;
    description
    "Total number of update events sent to all the connected
    clients.";
  }
  leaf num-event-max {
    type yang:counter32;
    config false;
    description
    "The maximum number of update events sent to the connected
    clients.";
  }
  leaf num-event-min {
    type yang:counter32;
    config false;
    description
    "The minimum number of update events sent to the connected
    clients.";
  }
9. Security Considerations

TBD.

10. IANA Considerations

This document registers two URIs in the "IETF XML Registry" [RFC3688]. Following the format in RFC 3688, the following registrations are requested.

Registrant Contact: The IESG.
XML: N/A; the requested URI is an XML namespace.

Registrant Contact: The IESG.
XML: N/A; the requested URI is an XML namespace.

This document registers two YANG modules in the "YANG Module Names" registry [RFC6020].

Name: ietf-alto
Prefix: alto
Reference: [RFCthis]

Name: ietf-alto-stats
Prefix: alto
Reference: [RFCthis]

[RFC Editor: Please replace RFCthis with the published RFC number for this document.]

11. References
11.1. Normative References


11.2. Informative References


The base data model defined by ietf-alto.yang does not include any choice cases for information resource creation algorithms. But developers may augment the ietf-alto.yang data model with definitions for any custom creation algorithms for different information resources. The following example module demonstrates the parameters of a network map creation algorithm that translates an IETF layer 3 unicast topology into a network map.
module example-ietf-alto-alg {
    namespace "urn:example:ietf-alto-alg";
    prefix "alto-alg";

    import ietf-alto {
        prefix "alto";
    }

    augment "//alto:alto-server/alto:resource/alto:resource-params"
        + "//alto:networkmap/alto:alto-networkmap-params"
        + "//alto:algorithm" {
        case l3-unicast-cluster {
            container l3-unicast-cluster-algorithm {
                leaf l3-unicast-topo {
                    type leafref {
                        path "//alto:alto-server/data-source/source-id";
                    }
                    mandatory true;
                    description
                        "The data source to an IETF layer 3 unicast topology.";
                }
                leaf depth {
                    type uint32;
                    description
                        "The depth of the clustering.";
                }
            }
        }
    }
}

Acknowledgements

The authors thank Qiufang Ma and Qin Wu for their help with drafting
the initial version of the YANG modules. Thanks also to Adrian
Farrel, Qiao Xiang, Qin Wu, and Qiufang Ma for their reviews and
valuable feedback.

Authors' Addresses

Jingxuan Jensen Zhang
Tongji University
4800 Cao’An Hwy
Shanghai
201804
China
Email: jingxuan.n.zhang@gmail.com
Manufacturer Usage Description (MUD) (D)TLS Profiles for IoT Devices
draft-ietf-opsawg-mud-tls-06

Abstract

This memo extends the Manufacturer Usage Description (MUD) specification to incorporate (D)TLS profile parameters. This allows a network security service to identify unexpected (D)TLS usage, which can indicate the presence of unauthorized software or malware on an endpoint.

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 6 September 2022.

Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.
1. Introduction

Encryption is necessary to enhance the privacy of end users using IoT devices. TLS [RFC8446] and DTLS [I-D.ietf-tls-dtls13] are the dominant protocols (counting all (D)TLS versions) providing encryption for IoT device traffic. Unfortunately, in conjunction with IoT applications’ rise of encryption, malware authors are also using encryption which thwarts network-based analysis such as deep packet inspection (DPI). Other mechanisms are thus needed to help
detecting malware running on an IoT device.

Malware frequently uses proprietary libraries for its activities, and those libraries are reused much like any other software engineering project. [malware] indicates that there are observable differences in how malware uses encryption compared with how non-malware uses encryption. There are several interesting findings specific to (D)TLS which were found common to malware:

* Older and weaker cryptographic parameters (e.g., TLS_RSA_WITH_RC4_128_SHA).

* TLS server name indication (SNI) extension [RFC6066] and server certificates are composed of subjects with characteristics of a domain generation algorithm (DGA) (e.g., ‘www.33mhwt2j.net’).

* Higher use of self-signed certificates compared with typical legitimate software.

* Discrepancies in the SNI TLS extension and the DNS names in the SubjectAltName (SAN) X.509 extension in the server certificate message.

* Discrepancies in the key exchange algorithm and the client public key length in comparison with legitimate flows. As a reminder, the Client Key Exchange message has been removed from TLS 1.3.

* Lower diversity in TLS client advertised extensions compared to legitimate clients.

* Using privacy enhancing technologies like Tor, Psiphon, Ultrasurf (see [malware-tls]), and evasion techniques such as ClientHello randomization.

* Using DNS-over-HTTPS (DoH) [RFC8484] to avoid detection by malware DNS filtering services [malware-doh]. Specifically, malware may not use the DoH server provided by the local network.

If observable (D)TLS profile parameters are used, the following functions are possible which have a positive impact on the local network security:

* Permit intended DTLS or TLS use and block malicious DTLS or TLS use. This is superior to the layers 3 and 4 ACLs of Manufacturer Usage Description Specification (MUD) [RFC8520] which are not suitable for broad communication patterns.
* Ensure TLS certificates are valid. Several TLS deployments have been vulnerable to active Man-In-The-Middle (MITM) attacks because of the lack of certificate validation or vulnerability in the certificate validation function (see [crypto-vulnerability]). By observing (D)TLS profile parameters, a network element can detect when the TLS SNI mismatches the SubjectAltName and when the server’s certificate is invalid. In TLS 1.2, the ClientHello, ServerHello and Certificate messages are all sent in clear-text. This check is not possible with TLS 1.3, which encrypts the Certificate message thereby hiding the server identity from any intermediary. In TLS 1.3, the server certificate validation functions should be executed within an on-path TLS proxy, if such a proxy exists.

* Support new communication patterns. An IoT device can learn a new capability, and the new capability can change the way the IoT device communicates with other devices located in the local network and Internet. There would be an inaccurate policy if an IoT device rapidly changes the IP addresses and domain names it communicates with while the MUD ACLs were slower to update (see [clear-as-mud]). In such a case, observable (D)TLS profile parameters can be used to permit intended use and to block malicious behavior from the IoT device.

The YANG module specified in Section 5 of this document is an extension of YANG Data Model for Network Access Control Lists (ACLs) [RFC8519] to enhance MUD [RFC8520] to model observable (D)TLS profile parameters. Using these (D)TLS profile parameters, an active MUD-enforcing network security service (e.g., firewall) can identify MUD non-compliant (D)TLS behavior indicating outdated cryptography or malware. This detection can prevent malware downloads, block access to malicious domains, enforce use of strong ciphers, stop data exfiltration, etc. In addition, organizations may have policies around acceptable ciphers and certificates for the websites the IoT devices connect to. Examples include no use of old and less secure versions of TLS, no use of self-signed certificates, deny-list or accept-list of Certificate Authorities, valid certificate expiration time, etc. These policies can be enforced by observing the (D)TLS profile parameters. Network security services can use the IoT device’s (D)TLS profile parameters to identify legitimate flows by observing (D)TLS sessions, and can make inferences to permit legitimate flows and to block malicious or insecure flows. The proposed technique is also suitable in deployments where decryption techniques are not ideal due to privacy concerns, non-cooperating end-points, and expense.
2. Terminology

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

"(D)TLS" is used for statements that apply to both Transport Layer Security [RFC8446] and Datagram Transport Layer Security [RFC6347]. Specific terms are used for any statement that applies to either protocol alone.

'DoH/DoT' refers to DNS-over-HTTPS and/or DNS-over-TLS.

3. Overview of MUD (D)TLS profiles for IoT devices

In Enterprise networks, protection and detection are typically done both on end hosts and in the network. Host security agents have deep visibility on the devices where they are installed, whereas the network has broader visibility. Installing host security agents may not be a viable option on IoT devices, and network-based security is an efficient means to protect such IoT devices. If the IoT device supports a MUD (D)TLS profile, the (D)TLS profile parameters of the IoT device can be used by a middlebox to detect and block malware communication, while at the same time preserving the privacy of legitimate uses of encryption. The middlebox need not proxy (D)TLS but can passively observe the parameters of (D)TLS handshakes from IoT devices and gain visibility into TLS 1.2 parameters and partial visibility into TLS 1.3 parameters.
Malicious agents can try to use the (D)TLS profile parameters of legitimate agents to evade detection, but it becomes a challenge to mimic the behavior of various IoT device types and IoT device models from several manufacturers. In other words, malware developers will have to develop malicious agents per IoT device type, manufacturer and model, infect the device with the tailored malware agent and will have keep up with updates to the device’s (D)TLS profile parameters over time. Furthermore, the malware’s command and control server certificates need to be signed by the same certifying authorities trusted by the IoT devices. Typically, IoT devices have an infrastructure that supports a rapid deployment of updates, and malware agents will have a near-impossible task of similarly deploying updates and continuing to mimic the TLS behavior of the IoT device it has infected. However, if the IoT device has reached end-of-life and the IoT manufacturer will not issue a firmware or software update to the Thing or will not update the MUD file, the "is-supported" attribute defined in Section 3.6 of [RFC8520] can be used by the MUD manager to identify the IoT manufacturer no longer supports the device.

The end-of-life of a device does not necessarily mean that it is defective; rather, it denotes a need to replace and upgrade the network to next-generation devices for additional functionality. The network security service will have to rely on other techniques discussed in Section 8 to identify malicious connections until the device is replaced.

Compromised IoT devices are typically used for launching DDoS attacks (Section 3 of [RFC8576]). For example, DDoS attacks like Slowloris and Transport Layer Security (TLS) re-negotiation can be blocked if the victim’s server certificate is not be signed by the same certifying authorities trusted by the IoT device.

4. (D)TLS 1.3 Handshake

In (D)TLS 1.3, full (D)TLS handshake inspection is not possible since all (D)TLS handshake messages excluding the ClientHello message are encrypted. (D)TLS 1.3 has introduced new extensions in the handshake record layers called Encrypted Extensions. Using these extensions handshake messages will be encrypted and network security services (such as a firewall) are incapable to decipher the handshake, and thus cannot view the server certificate. However, the ClientHello and ServerHello still have some fields visible, such as the list of supported versions, named groups, cipher suites, signature algorithms and extensions in ClientHello, and chosen cipher in the ServerHello. For instance, if the malware uses evasion techniques like ClientHello randomization, the observable list of cipher suites and extensions offered by the malware agent in the ClientHello message will not
match the list of cipher suites and extensions offered by the legitimate client in the ClientHello message, and the middlebox can block malicious flows without acting as a (D)TLS 1.3 proxy.

4.1. Full (D)TLS 1.3 Handshake Inspection

To obtain more visibility into negotiated TLS 1.3 parameters, a middlebox can act as a (D)TLS 1.3 proxy. A middlebox can act as a (D)TLS proxy for the IoT devices owned and managed by the IT team in the Enterprise network and the (D)TLS proxy must meet the security and privacy requirements of the organization. In other words, the scope of middlebox acting as a (D)TLS proxy is restricted to Enterprise network owning and managing the IoT devices. The middlebox would have to follow the behaviour detailed in Section 9.3 of [RFC8446] to act as a compliant (D)TLS 1.3 proxy.

To further increase privacy, Encrypted Client Hello (ECH) extension [I-D.ietf-tls-esni] prevents passive observation of the TLS Server Name Indication extension and other potentially sensitive fields, such as the ALPN [RFC7301]. To effectively provide that privacy protection, ECH extension needs to be used in conjunction with DNSSEC or DNS encryption (e.g., DoH). A middlebox (e.g., firewall) passively inspecting ECH extension cannot observe the encrypted SNI nor observe the encrypted DNS traffic. The middlebox would have to follow the behaviour detailed in [I-D.ietf-tls-esni] to disable ECH or fake ECH records in the DNS response so that the client encrypts data to the middlebox or strips the ECH record from the DNS response. However, if the client performs full DNSSEC validation locally [RFC6698], it can detect forged DNS responses.

4.2. Encrypted DNS

A common usage pattern for certain type of IoT devices (e.g., light bulb) is for it to "call home" to a service that resides on the public Internet, where that service is referenced through a domain name (A or AAAA record). As discussed in Manufacturer Usage Description Specification [RFC8520], because these devices tend to require access to very few sites, all other access should be considered suspect. If an IoT device is pre-configured to use a DNS resolver not signaled by the network, the MUD policy enforcement point is moved to that resolver, which cannot enforce the MUD policy based on domain names (Section 8 of [RFC8520]). If the DNS query is not accessible for inspection, it becomes quite difficult for the infrastructure to suspect anything. Thus the use of a DNS resolver not signaled by the network is incompatible with MUD in general. A network-designated DoH/DoT server is necessary to allow MUD policy enforcement on the local network (Section 6.5 of [I-D.ietf-opsawg-mud-iot-dns-considerations]).
5. (D)TLS Profile of a IoT device

This document specifies a YANG module for representing (D)TLS profile. The (D)TLS profile YANG module provides a method for network security services to observe the (D)TLS profile parameters in the (D)TLS handshake to permit intended use and to block malicious behavior. This module uses the cryptographic types defined in [I-D.ietf-netconf-crypto-types]. See [RFC7925] for (D)TLS 1.2 and [I-D.ietf-uta-tls13-iot-profile] for DTLS 1.3 recommendations related to IoT devices, and [RFC7525] for additional (D)TLS 1.2 recommendations.

A companion YANG module is defined to include a collection of (D)TLS parameters and (D)TLS versions maintained by IANA: "iana-tls-profile" (Section 5.3).

The (D)TLS parameters in each (D)TLS profile include the following:

* Profile name
* (D)TLS versions supported by the IoT device.
* List of supported cipher suites. For (D)TLS1.2, [RFC7925] recommends AEAD ciphers for IoT devices.
* List of supported extension types
* List of trust anchor certificates used by the IoT device. If the server certificate is signed by one of the trust anchors, the middlebox continues with the connection as normal. Otherwise, the middlebox will react as if the server certificate validation has failed and takes appropriate action (e.g., block the (D)TLS session). An IoT device can use a private trust anchor to validate a server’s certificate (e.g., the private trust anchor can be preloaded at manufacturing time on the IoT device and the IoT device fetches the firmware image from the Firmware server whose certificate is signed by the private CA). This empowers the middlebox to reject TLS sessions to servers that the IoT device does not trust.
* List of SPKI pin set pre-configured on the client to validate self-signed server certificates or raw public keys. A SPKI pin set is a cryptographic digest to "pin" public key information in a manner similar to HTTP Public Key Pinning (HPKP) [RFC7469]. If SPKI pin set is present in the (D)TLS profile of a IoT device and the server certificate does not pass the PKIX certification path validation, the middlebox computes the SPKI Fingerprint for the public key found in the server’s certificate (or in the raw public...
key, if the server provides that instead). If a computed fingerprint exactly matches one of the SPKI pin sets in the (D)TLS profile, the middlebox continues with the connection as normal. Otherwise, the middlebox will act on the SPKI validation failure and take appropriate action.

* Cryptographic hash algorithm used to generate the SPKI pinsets

* List of pre-shared key exchange modes

* List of named groups (DHE or ECDHE) supported by the client

* List of signature algorithms the client can validate in X.509 server certificates

* List of signature algorithms the client is willing to accept for CertificateVerify message (Section 4.2.3 of [RFC8446]). For example, a TLS client implementation can support different sets of algorithms for certificates and in TLS to signal the capabilities in "signature_algorithms_cert" and "signature_algorithms" extensions.

* List of supported application protocols (e.g., h3, h2, http/1.1 etc.)

* List of certificate compression algorithms (defined in [I-D.ietf-tls-certificate-compression])

* List of the distinguished names [X501] of acceptable certificate authorities, represented in DER-encoded format [X690] (defined in Section 4.2.4 of [RFC8446])

GREASE [RFC8701] sends random values on TLS parameters to ensure future extensibility of TLS extensions. Similar random values might be extended to other TLS parameters. Thus, the (D)TLS profile parameters defined in the YANG module by this document MUST NOT include the GREASE values for extension types, named groups, signature algorithms, (D)TLS versions, pre-shared key exchange modes, cipher suites and for any other TLS parameters defined in future RFCs.

The (D)TLS profile does not include parameters like compression methods for data compression, [RFC7525] recommends disabling TLS-level compression to prevent compression-related attacks. In TLS 1.3, only the "null" compression method is allowed (Section 4.1.2 of [RFC8446]).
5.1. Tree Structure of the (D)TLS profile Extension to the ACL YANG Model

This document augments the "ietf-acl" ACL YANG module defined in [RFC8519] for signaling the IoT device (D)TLS profile. This document defines the YANG module "ietf-acl-tls", which has the following tree structure:

```
module: ietf-acl-tls
    augment /acl:acls/acl:acl/acl:aces/acl:ace/acl:matches:
        +++rw client-profile (match-on-tls-dtls)?
        +++rw tls-dtls-profiles* [profile-name]
            +++w profile-name string
            +++w supported-tls-versions* ianatp:tls-version
            +++w supported-dtls-versions* ianatp:dtls-version
            +++w cipher-suites* [cipher hash]
                +++w cipher ianatp:cipher-algorithm
                +++w hash ianatp:hash-algorithm
            +++w extension-types*
                ianatp:extension-type
            +++w acceptlist-ta-certs*  
                ct:trust-anchor-cert-cms
            +++w spki
                +++w spki-pin-sets* ianatp:spki-pin-set
                +++w spki-hash-algorithm? iha:hash-algorithm-type
            +++w psk-key-exchange-modes*
                ianatp:psk-key-exchange-mode
                {tls-1-3 or dtls-1-3}?
            +++w supported-groups*  
                ianatp:supported-group
            +++w signature-algorithms-cert*
                ianatp:signature-algorithm
                {tls-1-3 or dtls-1-3}?
            +++w signature-algorithms*  
                ianatp:signature-algorithm
            +++w application-protocols*
                ianatp:application-protocol
            +++w cert-compression-algorithms*
                ianatp:cert-compression-algorithm
                {tls-1-3 or dtls-1-3}?
            +++w certificate-authorities*
                ianatp:certificate-authority
                {tls-1-3 or dtls-1-3}?
```

5.2. The (D)TLS profile Extension to the ACL YANG Model
<CODE BEGINS> file "ietf-acl-tls@2020-10-07.yang"
module ietf-acl-tls {
    yang-version 1.1;
    prefix ietf-acl-tls;

    import iana-tls-profile {
        prefix ianatp;
        reference
            "RFC XXXX: Manufacturer Usage Description (MUD) (D)TLS Profiles for IoT Devices";
    }
    import ietf-crypto-types {
        prefix ct;
        reference
            "RFC CCCC: Common YANG Data Types for Cryptography";
    }
    import iana-hash-algs {
        prefix iha;
        reference
            "RFC IIII: Common YANG Data Types for Hash algorithms";
    }
    import ietf-access-control-list {
        prefix acl;
        reference
            "RFC 8519: YANG Data Model for Network Access Control Lists (ACLs)";
    }

    organization
        "IETF OPSAWG (Operations and Management Area Working Group)";
    contact
        "WG Web: <https://datatracker.ietf.org/wg/opsawg/>"
        "WG List: opsawg@ietf.org"
        "Author: Konda, Tirumaleswar Reddy"
        "TirumaleswarReddy_Konda@McAfee.com"
    description
        "This YANG module defines a component that augments the IETF description of an access list to allow (D)TLS profile as matching criteria."

    Copyright (c) 2020 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 Simplified BSD License
set forth in Section 4.c of the IETF Trust's Legal Provisions
Relating to IETF Documents
(http://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 2020-11-02 {
  description
    "Initial revision";
  reference
    "RFC XXXX: Manufacturer Usage Description (MUD) (D)TLS
    Profiles for IoT Devices";
}

feature tls-1-2 {
  description
    "TLS Protocol Version 1.2 is supported.";
  reference
    Version 1.2";
}

feature tls-1-3 {
  description
    "TLS Protocol Version 1.3 is supported.";
  reference
    Version 1.3";
}

feature dtls-1-2 {
  description
    "DTLS Protocol Version 1.2 is supported.";
  reference
    "RFC 6346: Datagram Transport Layer Security
    Version 1.2";
}

feature dtls-1-3 {
  description
    "DTLS Protocol Version 1.3 is supported.";
  reference
    "draft-ietf-tls-dtls13: Datagram Transport Layer
    Security 1.3";
}
feature match-on-tls-dtls {
    description
        "The networking device can support matching on
        (D)TLS parameters.";
}

    if-feature "match-on-tls-dtls";
    description
        "(D)TLS specific matches."
} container client-profile {
    description
        "A grouping for (D)TLS profiles.";
} container client-profile {
    description
        "A grouping for DTLS profiles.";
} list tls-dtls-profiles {
    key "profile-name";
    description
        "A list of (D)TLS version profiles supported by
        the client.";
} leaf profile-name {
    type string {
        length "1..64";
    }
    description
        "The name of (D)TLS profile; space and special
        characters are not allowed.";
} leaf-list supported-tls-versions {
    type ianatp:tls-version;
    description
        "TLS versions supported by the client.";
} leaf-list supported-dtls-versions {
    type ianatp:dtls-version;
    description
        "DTLS versions supported by the client.";
} list cipher-suites {
    key "cipher hash";
    leaf cipher {
        type ianatp:cipher-algorithm;
        description
            "AEAD encryption algorithm as defined in RFC5116.";
    }
    leaf hash {
        type ianatp:hash-algorithm;
description
"Hash algorithm used with HKDF as defined in RFC5869.";
}
description
"A list of Cipher Suites supported by the client."
}
leaf-list extension-types {
  type ianatp:extension-type;
description
  "A list of Extension Types supported by the client."
}
leaf-list acceptlist-ta-certs {
  type ct:trust-anchor-cert-cms;
description
  "A list of trust anchor certificates used by the client."
}
container spki {
  description
  "A grouping for spki."
  leaf-list spki-pin-sets {
    type ianatp:spki-pin-set;
description
    "A list of SPKI pin sets pre-configured on the client
to validate self-signed server certificate or
raw public key."
  }
  leaf spki-hash-algorithm {
    type iha:hash-algorithm-type;
description
    "cryptographic hash algorithm used to generate the
SPKI pinset."
  }
}
leaf-list psk-key-exchange-modes {
  if-feature "tls-1-3 or dtls-1-3";
type ianatp:psk-key-exchange-mode;
description
  "pre-shared key exchange modes."
}
leaf-list supported-groups {
  type ianatp:supported-group;
description
  "A list of named groups supported by the client."
}
leaf-list signature-algorithms-cert {
  if-feature "tls-1-3 or dtls-1-3";
type ianatp:signature-algorithm;
description
"A list signature algorithms the client can validate in X.509 certificates."
)
leaf-list signature-algorithms {
  type ianatp:signature-algorithm;
  description
  "A list signature algorithms the client can validate in the CertificateVerify message.";
}
leaf-list application-protocols {
  type ianatp:application-protocol;
  description
  "A list application protocols supported by the client.";
}
leaf-list cert-compression-algorithms {
  if-feature "tls-1-3 or dtls-1-3";
  type ianatp:cert-compression-algorithm;
  description
  "A list certificate compression algorithms supported by the client.";
}
leaf-list certificateAuthorities {
  if-feature "tls-1-3 or dtls-1-3";
  type ianatp:certificate-authority;
  description
  "A list of the distinguished names of certificate authorities acceptable to the client.";
}

5.3. IANA (D)TLS profile YANG Module

The TLS and DTLS IANA registries are available from https://www.iana.org/assignments/tls-parameters/tls-parameters.txt and https://www.iana.org/assignments/tls-extensiontype-values/tls-extensiontype-values.txt.

The values for all the parameters in the "iana-tls-profile" YANG module are defined in the TLS and DTLS IANA registries excluding the tls-version, dtls-version, spki-pin-set, and certificate-authority parameters. The values of spki-pin-set and certificate-authority parameters will be specific to the IoT device.
The TLS and DTLS IANA registries do not maintain (D)TLS version numbers. In (D)TLS 1.2 and below, "legacy_version" field in the ClientHello message is used for version negotiation. However in (D)TLS 1.3, the "supported_versions" extension is used by the client to indicate which versions of (D)TLS it supports. TLS 1.3 ClientHello messages are identified as having a "legacy_version" of 0x0303 and a "supported_versions" extension present with 0x0304 as the highest version. DTLS 1.3 ClientHello messages are identified as having a "legacy_version" of 0xfefd and a "supported_versions" extension present with 0x0304 as the highest version.

In order to ease updating the "iana-tls-profile" YANG module with future (D)TLS versions, new (D)TLS version registries are defined in Section 10.2 and Section 10.3. Whenever a new (D)TLS protocol version is defined, the registry will be updated using expert review; the "iana-tls-profile" YANG module will be automatically updated by IANA.

The "iana-tls-profile" YANG module is defined as follows:

<CODE BEGINS> file "iana-tls-profile@2020-10-07.yang"
module iana-tls-profile { 
    yang-version 1.1;
    namespace "urn:ietf:params:xml:ns:yang:iana-tls-profile";
    prefix ianatp;

    organization
        "IANA";
    contact
        "Internet Assigned Numbers Authority"
        "Postal: ICANN
12025 Waterfront Drive, Suite 300
Los Angeles, CA  90094-2536
United States"
        "Tel: +1 310 301 5800
E-Mail: iana@iana.org"; 
    description
        "This module contains YANG definition for the (D)TLS profile."

    Copyright (c) 2020 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 Simplified BSD License set forth in Section 4.c of the IETF Trust’s Legal Provisions
typedef extension-type {
  type uint16;
  description
    "Extension type in the TLS ExtensionType Values registry as
      defined in Section 7 of RFC8447.";
}

typedef supported-group {
  type uint16;
  description
    "Supported Group in the TLS Supported Groups registry as
      defined in Section 9 of RFC8447.";
}

typedef spki-pin-set {
  type binary;
  description
    "Subject Public Key Info pin set as discussed in
      Section 2.4 of RFC7469.";
}

typedef signature-algorithm {
  type uint16;
  description
    "Signature algorithm in the TLS SignatureScheme registry as
      defined in Section 11 of RFC8446.";
}

typedef psk-key-exchange-mode {
  type uint8;
  description
    "Pre-shared key exchange mode in the TLS PskKeyExchangeMode
      registry as defined in Section 11 of RFC8446.";
}
typedef application-protocol {
  type string;
  description
    "Application-Layer Protocol Negotiation (ALPN) Protocol ID
    registry as defined in Section 6 of RFC7301.";
}

typedef cert-compression-algorithm {
  type uint16;
  description
    "Certificate compression algorithm in TLS Certificate
    Compression Algorithm IDs registry as defined in
    Section 7.3 of ietf-tls-certificate-compression";
}

typedef certificate-authority {
  type string;
  description
    "Distinguished Name of Certificate authority as discussed
    in Section 4.2.4 of RFC8446.";
}

typedef cipher-algorithm {
  type uint8;
  description
    "AEAD encryption algorithm in TLS Cipher Suites registry
    as discussed in Section 11 of RFC8446.";
}

typedef hash-algorithm {
  type uint8;
  description
    "Hash algorithm used with HMAC-based Extract-and-Expand Key
    Derivation Function (HKDF) in TLS Cipher Suites registry
    as discussed in Section 11 of RFC8446.";
}

typedef tls-version {
  type enumeration {
    enum tls-1.2 {
      value 1;
      description
        "TLS Protocol Version 1.2.
        TLS 1.2 ClientHello contains
        0x0303 in 'legacy_version'."
    }
    reference
    }
  }
}
Version 1.2;

enum tls-1.3 {
    value 2;
    description
        "TLS Protocol Version 1.3.
        TLS 1.3 ClientHello contains a
        supported_versions extension with 0x0304
        contained in its body and the ClientHello contains
        0x0303 in 'legacy_version'."
    reference
         Version 1.3";
}

description
    "Indicates the TLS version.";
}

typedef dtls-version {
    type enumeration {
        enum dtls-1.2 {
            value 1;
            description
                "DTLS Protocol Version 1.2.
                DTLS 1.2 ClientHello contains
                0xfefd in 'legacy_version'."
            reference
                "RFC 6346: Datagram Transport Layer Security 1.2";
        }
        enum dtls-1.3 {
            value 2;
            description
                "DTLS Protocol Version 1.3.
                DTLS 1.3 ClientHello contains a
                supported_versions extension with 0x0304
                contained in its body and the ClientHello contains
                0xfefd in 'legacy_version'."
            reference
                "RFC DDDD: Datagram Transport Layer Security 1.3";
        }
    }
}
description
    "Indicates the DTLS version.";
5.4.  MUD (D)TLS Profile Extension

This document augments the "ietf-mud" MUD YANG module to indicate whether the device supports (D)TLS profile. If the "ietf-mud-tls" extension is supported by the device, MUD file is assumed to implement the "match-on-tls-dtls" ACL model feature defined in this specification. Furthermore, only "accept" or "drop" actions SHOULD be included with the (D)TLS profile similar to the actions allowed in Section 2 of [RFC8520].

This document defines the YANG module "ietf-mud-tls", which has the following tree structure:

module: ietf-mud-tls
    augment /ietf-mud:mud:
        +--rw is-tls-dtls-profile-supported?   boolean

The model is defined as follows:

<CODE BEGINS> file "iana-tls-mud@2020-10-20.yang"
module ietf-mud-tls {
    yang-version 1.1;
    prefix ietf-mud-tls;

    import ietf-mud {
        prefix ietf-mud;
    }

    organization
        "IETF OPSAWG (Operations and Management Area Working Group)"
    contact
        "WG Web: <https://datatracker.ietf.org/wg/opsawg/>
        WG List: opsawg@ietf.org

    Author: Konda, Tirumaleswar Reddy
        TirumaleswarReddy_Konda@McAfee.com
        ";
    description
        "Extension to a MUD module to indicate (D)TLS profile support."

    Copyright (c) 2020 IETF Trust and the persons identified as authors of the code. All rights reserved.

6. Processing of the MUD (D)TLS Profile

The following text outlines the rules for a network security service (e.g., firewall) to follow to process the MUD (D)TLS Profile:

* If the (D)TLS parameter observed in a (D)TLS session is not specified in the MUD (D)TLS profile and the parameter is recognized by the firewall, it can identify unexpected (D)TLS usage, which can indicate the presence of unauthorized software or malware on an endpoint. The firewall can take several actions like block the (D)TLS session or raise an alert to quarantine and remediate the compromised device. For example, if the cipher suite TLS_RSA_WITH_AES_128_CBC_SHA in the ClientHello message is not specified in the MUD (D)TLS profile and the cipher suite is recognized by the firewall, it can identify unexpected TLS usage.
* If the (D)TLS parameter observed in a (D)TLS session is not specified in the MUD (D)TLS profile and the (D)TLS parameter is not recognized by the firewall, it can ignore the unrecognized parameter and the correct behavior is not to block the (D)TLS session. The behavior is functionally equivalent to the compliant TLS middlebox description in Section 9.3 of [RFC8446] to ignore all unrecognized cipher suites, extensions, and other parameters. For example, if the cipher suite TLS_CHACHA20_POLY1305_SHA256 in the ClientHello message is not specified in the MUD (D)TLS profile and the cipher suite is not recognized by the firewall, it can ignore the unrecognized cipher suite.

* Deployments update at different rates, so an updated MUD (D)TLS profile may support newer parameters. If the firewall does not recognize the newer parameters, an alert should be triggered to the firewall vendor and the IoT device owner or administrator. A firewall must be readily updatable, so that when new parameters in the MUD (D)TLS profile are discovered that are not recognized by the firewall, it can be updated quickly. Most importantly, if the firewall is not readily updatable, its protection efficacy to identify emerging malware will decrease with time. For example, if the cipher suite TLS_AES_128_CCM_8_SHA256 specified in the MUD (D)TLS profile is not recognized by the firewall, an alert will be triggered. Similarly, if the (D)TLS version specified in the MUD file is not recognized by the firewall, an alert will be triggered.

7. MUD File Example

The example below contains (D)TLS profile parameters for a IoT device used to reach servers listening on port 443 using TCP transport. JSON encoding of YANG modelled data [RFC7951] is used to illustrate the example.

```json
{
  "ietf-mud:mud": {
    "mud-version": 1,
    "mud-url": "https://example.com/IoTDevice",
    "last-update": "2019-18-06T03:56:40.105+10:00",
    "cache-validity": 100,
    "extensions": [
      "ietf-mud-tls"
    ],
    "ietf-mud-tls:is-tls-dtls-profile-supported": "true",
    "is-supported": true,
    "systeminfo": "IoT device name",
    "from-device-policy": {
```
"access-lists": {
"access-list": [
{
"name": "mud-7500-profile"
}
]
},
"ietf-access-control-list:acls": {
"acl": [
{
"name": "mud-7500-profile",
"type": "ipv6-acl-type",
"aces": {
"ace": [
{
"name": "cl0-frdev",
"matches": {
"ipv6": {
"protocol": 6
},
"tcp": {
"ietf-mud:direction-initiated": "from-device",
"destination-port": {
"operator": "eq",
"port": 443
}
},
"ietf-acl-tls:client-profile": {
"tls-dtls-profiles": [
{
"supported-tls-versions": ["tls-1.3"],
"cipher-suites": [
{
"cipher": 19,
"hash": 1
},
{ "cipher": 19,
"hash": 2
}
],
"extension-types": [10,11,13,16,24],
"supported-groups": [29]
}
]}
},
"actions": {

The following illustrates the example scenarios for processing the above profile:

* If the extension type "encrypt_then_mac" (code point 22) [RFC7366] in the ClientHello message is recognized by the firewall, it can identify unexpected TLS usage.

* If the extension type "token_binding" (code point 24) [RFC8472] in the MUD (D)TLS profile is not recognized by the firewall, it can ignore the unrecognized extension. Because the extension type "token_binding" is specified in the profile, an alert will be triggered to the firewall vendor and the IoT device owner or administrator to notify the firewall is not up to date.

8. Security Considerations

Security considerations in [RFC8520] need to be taken into consideration. The middlebox must adhere to the invariants discussed in Section 9.3 of [RFC8446] to act as a compliant proxy.

Although it is challenging for a malware to mimic the TLS behavior of various IoT device types and IoT device models from several manufacturers, malicious agents have a very low probability of using the same (D)TLS profile parameters as legitimate agents on the IoT device to evade detection. Network security services should also rely on contextual network data to detect false negatives. In order to detect such malicious flows, anomaly detection (deep learning techniques on network data) can be used to detect malicious agents using the same (D)TLS profile parameters as legitimate agent on the IoT device. In anomaly detection, the main idea is to maintain rigorous learning of "normal" behavior and where an "anomaly" (or an attack) is identified and categorized based on the knowledge about the normal behavior and a deviation from this normal behavior.
9. Privacy Considerations

Privacy considerations discussed in Section 16 of [RFC8520] to not reveal the MUD URL to an attacker need to be taken into consideration. The MUD URL can be stored in Trusted Execution Environment (TEE) for secure operation, enhanced data security, and prevent exposure to unauthorized software.

Full handshake inspection (Section 4.1) requires a TLS proxy device which needs to decrypt traffic between the IoT device and its server(s). There is a tradeoff between privacy of the data carried inside TLS (especially e.g., personally identifiable information and protected health information) and efficacy of endpoint security. It is strongly RECOMMENDED to avoid a TLS proxy whenever possible. For example, an enterprise firewall administrator can configure the middlebox to bypass TLS proxy functionality or payload inspection for connections destined to specific well-known services. Alternatively, a IoT device could be configured to reject all sessions that involve proxy servers to specific well-known services. In addition, mechanisms based on object security can be used by IoT devices to enable end-to-end security and the middlebox will not have any access to the packet data. For example, Object Security for Constrained RESTful Environments (OSCORE) [RFC8613] is a proposal that protects CoAP messages by wrapping them in the COSE format [RFC8152].

10. IANA Considerations

10.1. (D)TLS Profile YANG Modules

This document requests IANA to register the following URIs in the "ns" subregistry within the "IETF XML Registry" [RFC3688]:

Registrant Contact: The IESG.
XML: N/A; the requested URI is an XML namespace.

Registrant Contact: The IESG.
XML: N/A; the requested URI is an XML namespace.

Registrant Contact: The IESG.
XML: N/A; the requested URI is an XML namespace.
IANA is requested to create an IANA-maintained YANG Module called "iana-tls-profile", based on the contents of Section 5.3, which will allow for new (D)TLS parameters and (D)TLS versions to be added to "client-profile". The registration procedure will be Expert Review, as defined by [RFC8126].

This document requests IANA to register the following YANG modules in the "YANG Module Names" subregistry [RFC6020] within the "YANG Parameters" registry.

name: iana-tls-profile
maintained by IANA: Y
prefix: ianatp
reference: RFC XXXX

name: ietf-acl-tls
maintained by IANA: N
prefix: ietf-acl-tls
reference: RFC XXXX

name: ietf-mud-tls
maintained by IANA: N
prefix: ietf-mud-tls
reference: RFC XXXX

IANA is requested to create an initial version of the IANA-maintained YANG Module called "iana-tls-profile", based on the contents of Section 5.3, which will allow for new (D)TLS parameters and (D)TLS versions to be added. IANA is requested to add this note:

* tls-version and dtls-version values must not be directly added to the iana-tls-profile YANG module. They must instead be respectively added to the "ACL TLS Version Codes", and "ACL DTLS Version Codes" registries.

* (D)TLS parameters must not be directly added to the iana-tls-profile YANG module. They must instead be added to the "ACL (D)TLS Parameters" registry.

When a 'tls-version' or 'dtls-version' value is respectively added to the "ACL TLS Version Codes" or "ACL DTLS Version Codes" registry, a new "enum" statement must be added to the iana-tls-profile YANG module. The following "enum" statement, and substatements thereof, should be defined:
"enum": Replicates the label from the registry.

"value": Contains the IANA-assigned value corresponding to the 'tls-version' or 'dtls-version'.

"description": Replicates the description from the registry.

"reference": Replicates the reference from the registry and adds the title of the document.

When a (D)TLS parameter is added to "ACL (D)TLS Parameters" registry, a new "type" statement must be added to the iana-tls-profile YANG module. The following "type" statement, and substatements thereof, should be defined:

"derived type": Replicates the parameter name from the registry.

"built-in type": Contains the built-in YANG type.

"description": Replicates the description from the registry.

When the iana-tls-profile YANG module is updated, a new "revision" statement must be added in front of the existing revision statements.

IANA is requested to add this note to "ACL TLS Version Codes", "ACL DTLS Version Codes", and "ACL (D)TLS Parameters" registries:

When this registry is modified, the YANG module iana-tls-profile must be updated as defined in [RFCXXXX].

The registration procedure for "ietf-acl-tls" YANG module will be Specification Required, as defined by [RFC8126].

10.2. ACL TLS Version registry

IANA is requested to create a new registry titled "ACL TLS Version Codes". Codes in this registry are used as valid values of 'tls-version' parameter. Further assignments are to be made through Expert Review [RFC8126].
<table>
<thead>
<tr>
<th>Value</th>
<th>Label</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>tls-1.2</td>
<td>TLS Version 1.2</td>
<td>[RFC5246]</td>
</tr>
<tr>
<td>2</td>
<td>tls-1.3</td>
<td>TLS Version 1.3</td>
<td>[RFC8446]</td>
</tr>
</tbody>
</table>

10.3. ACL DTLS version registry

IANA is requested to create a new registry titled "ACL DTLS Version Codes". Codes in this registry are used as valid values of 'dtls-version' parameter. Further assignments are to be made through Expert Review [RFC8126].

<table>
<thead>
<tr>
<th>Value</th>
<th>Label</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>dtls-1.2</td>
<td>DTLS Version 1.2</td>
<td>[RFC6346]</td>
</tr>
<tr>
<td>2</td>
<td>dtls-1.3</td>
<td>DTLS Version 1.3</td>
<td>[draft-ietf-tls-dtls13]</td>
</tr>
</tbody>
</table>

10.4. ACL (D)TLS Parameters registry

IANA is requested to create a new registry titled "ACL (D)TLS parameters".

The values for all the (D)TLS parameters in the registry are defined in the TLS and DTLS IANA registries (https://www.iana.org/assignments/tls-parameters/tls-parameters.txt and https://www.iana.org/assignments/tls-extensiontype-values/tls-extensiontype-values.txt) excluding the tls-version, dtls-version, spki-pin-set and certificate-authority parameters. Further assignments are to be made through Expert Review [RFC8126]. The registry is initially populated with the following parameters:
<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>YANG Type</th>
<th>JSON Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>extension-type</td>
<td>uint16</td>
<td>Number</td>
<td>Extension type</td>
</tr>
<tr>
<td>supported-group</td>
<td>uint16</td>
<td>Number</td>
<td>Supported group</td>
</tr>
<tr>
<td>spki-pin-set</td>
<td>binary</td>
<td>String</td>
<td>Subject public key info</td>
</tr>
<tr>
<td>signature-algorithm</td>
<td>uint16</td>
<td>Number</td>
<td>Signature algorithm</td>
</tr>
<tr>
<td>psk-key-exchange-mode</td>
<td>uint8</td>
<td>Number</td>
<td>pre-shared key exchange</td>
</tr>
<tr>
<td>application-protocol</td>
<td>string</td>
<td>String</td>
<td>Application protocol</td>
</tr>
<tr>
<td>cert-compression-algorithm</td>
<td>uint16</td>
<td>Number</td>
<td>Certificate compression</td>
</tr>
<tr>
<td>certificate-authority</td>
<td>string</td>
<td>String</td>
<td>Distinguished name of Cert Authority</td>
</tr>
<tr>
<td>cipher-algorithm</td>
<td>uint8</td>
<td>Number</td>
<td>AEAD encryption algorithm</td>
</tr>
<tr>
<td>hash-algorithm</td>
<td>uint8</td>
<td>Number</td>
<td>Hash algorithm</td>
</tr>
<tr>
<td>tls-version</td>
<td>enumeration</td>
<td>String</td>
<td>TLS version</td>
</tr>
<tr>
<td>dtls-version</td>
<td>enumeration</td>
<td>String</td>
<td>DTLS version</td>
</tr>
</tbody>
</table>
10.5. MUD Extensions registry

IANA is requested to create a new MUD Extension Name "ietf-mud-tls"
in the MUD Extensions IANA registry
https://www.iana.org/assignments/mud/mud.xhtml.

11. Acknowledgments

Thanks to Flemming Andreasen, Shashank Jain, Michael Richardson,
Piyush Joshi, Eliot Lear, Harsha Joshi, Qin Wu, Mohamed Boucadair,
Ben Schwartz, Eric Rescorla, Panwei William, Nick Lamb and Nick
Harper for the discussion and comments.

12. References

12.1. Normative References

[I-D.ietf-netconf-crypto-types]

[I-D.ietf-tls-certificate-compression]

[I-D.ietf-tls-dtls13]


12.2. Informative References

[cryto-vulnerability]

[I-D.ietf-opsawg-mud-iot-dns-considerations]

[I-D.ietf-tls-esni]

[I-D.ietf-uta-tls13-iot-profile]

[malware]


Authors' Addresses
Tirumaleswar Reddy
McAfee, Inc.
Embassy Golf Link Business Park
Bangalore 560071
Karnataka
India
Email: kondtir@gmail.com

Dan Wing
Citrix Systems, Inc.
4988 Great America Pkwy
Santa Clara, CA 95054
United States of America
Email: danwing@gmail.com

Blake Anderson
Cisco Systems, Inc.
170 West Tasman Dr
San Jose, CA 95134
United States of America
Email: blake.anderson@cisco.com
Abstract

This document proposes the Intra-Network eXposure analyzer Utility (INXU) as a vulnerability management solution for IoT networks. The goal of INXU is to take advantage of the functions of the RFC 8520 to allow a Security Experts Team on protecting multiple heterogeneous IoT networks, even when there is a few or none private information of the networks.

INXU identifies and analyzes the capability of an IoT device being exploited by an well known malicious activity. We also propose the Malicious Traffic Description (MTD), a data-model to describe traffic related to malicious activities.

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 27 November 2022.

Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.
This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction ................................................. 3
   1.1. Simple Example ....................................... 5
   1.2. Key Aspects .......................................... 5
   1.3. INXU Intended Use ..................................... 5
   1.4. Terminology .......................................... 6
2. The MTD Data Model ........................................... 7
   2.1. The draft-inxu-mtd YANG Module ....................... 8
   2.2. MTD Data Model Definition of Control Fields in the Root
       "mtd" Container ....................................... 10
       2.2.1. mtd-url .......................................... 10
       2.2.2. mtd-signature ................................... 10
       2.2.3. last-update ..................................... 10
       2.2.4. cache-validity ................................... 10
   2.3. MTD Data Model Definition of Traffic Description Fields in
       the Root "mtd" Container .............................. 10
       2.3.1. traffic-list ..................................... 11
       2.3.1.1. name .......................................... 11
       2.3.1.2. specific-devices ............................. 11
       2.3.2. malicious-descriptions ......................... 11
       2.3.2.1. name .......................................... 11
       2.3.2.2. specific-devices ............................. 11
       2.3.2.3. critical-acl-sets ........................... 11
       2.3.2.4. to-device-attacks ............................ 12
       2.3.2.5. from-device-attacks .......................... 12
       2.3.2.6. not-attack-traffic ........................... 12
   2.4. Augmentation to the ACL Model .......................... 12
       2.4.1. mtd:local-networks .............................. 12
       2.4.2. direction-initiated .............................. 12
       2.4.3. src-dnsname and dst-dnsname ....................... 13
       2.4.4. risk ............................................. 13
       2.4.5. risk-threshold .................................. 13
       2.4.6. alert-threshold .................................. 13
2.5. The MTD YANG Model ...................................... 13
2.6. MTD File Example ........................................ 21
3. The Intra-Network eXposure analyzer Utility .................. 29
   3.1. INXU Architecture and Components ..................... 30
   3.2. Workflow ............................................. 31
1. Introduction

While more Internet of Things (IoT) devices are deployed, the vulnerability management process turns even more difficult. This is mostly caused by the high heterogeneity and density of the IoT systems and devices, and challenges security teams on keeping Firewall and Intrusion Detection/Prevention Systems (IDS/IPS) rules up to date.

In some way, the Manufacturer Usage Description (MUD) [RFC8520] provides an alternative protection by reducing the capability of an IoT device being exploited -- as vector or target -- by malicious activities over the Internet. MUD does this by providing means to automatically build an allow-list of the IoT devices in a network based on the device manufacturers specification of expected traffic. This improves devices security by reducing their threat surface by blocking traffic with unexpected nodes/protocols, but still allows attacks which exploit vulnerabilities into the allowed traffic.

Besides this lack, the implementation of the [RFC8520] provides information that can support the identification of well-known vulnerabilities, as mentioned in its specification. This can be done by combining the allow-lists provided by the MUD manager into a communication graph of the connected IoT devices. With the communication graph, we can compare the traffic allowed by MUD with signatures of well-known malicious activities to identify -- and potentially block -- exposure of vulnerabilities into the network.
Integrating this analysis in traditional IDS or IPS can improve their efficacy and cover the MUD lack, but they only apply for scenarios where there is a security management team, such as corporate, smart grid and industrial IoT networks. On the other hand, in scenarios where there is a high heterogeneity of devices and low (or none) specialized support, such as in Home IoT Networks and Smart Cities, the process for keeping the attack signatures updated is not that simple.

Therefore, envisioning to overcome this gap, this document proposes INXU (Intra Network eXposure analyzer Utility) as a security tool that takes advantage of the MUD-based network communication graph to prevent the exploitation of well-known vulnerabilities. To do this, INXU blocks threats on the Local Area Network (LAN) after identifying them by comparing the signature of well-known malicious activities with the traffic flow allowed by the MUD. In short words, while MUD builds allow-lists, INXU builds a blocklist on top of MUD’s allow-lists.

The core component of INXU is Malicious Traffic Description (MTD), a document produced by a security specialist that describes ongoing malicious activities and well-known vulnerabilities and helps INXU find chains of connected IoT devices that can expose them to a threat. On top of MUD’s threat surface reduction, INXU adds another security layer that enables protection against incidents not addressed or even caused by the manufacturers.

The MTD data model, as in MUD, abstracts network addresses to allow describing the traffic without the need to know the network’s addressing schema or the connected devices. This resource allows creating portable descriptions of malicious traffic and protects the privacy in the LAN by not exposing private information to third parties in the security decision-making process. At the same time, it simplifies the sharing of knowledge about attacks between distinct networks.

Another relevant feature in INXU is its architecture that enables one Security Operation Center (SOC) to protect multiple distinct networks by sharing MTDs in a process similar to computer antivirus vaccines. This feature makes INXU a tool to protect LANs and the entire Internet ecosystem by making the operation of botnets and other attacks that affect the Internet’s stability more difficult.
1.1. Simple Example

An Internet Service Provider (ISP) connects tens to hundreds of houses to the Internet. Each one of these homes contains a wide range of IoT devices connected in their internal networks, in diverse topologies, and with different usages by each end-user. By the variety of scenarios, these home networks potentially contain a few devices infected by a DDoS capable botnet.

Due to the attacks carried by this botnet, frequently the ISP has a considerable part of its traffic being consumed by DDoS attacks, and often the clients call helpdesk for problems with devices caused by the botnet. The ISP knows that the malware’s infection occurs by a TCP/23 connection with a neighbour host, and the command and control occurs by a TCP/80 connection with a server located at mybotnet.example.com.

With this information, the ISP releases an MTD File describing this traffic, which can be used by its clients. In the home networks, the Customer Premises Equipment (CPE) collects the MTD File and compares it to the network communication graph provided by MUD, identifies exposures of vulnerabilities internally into the network, INXU evaluates the risk of the exposures and suggests blocks to prevent exploitations.

1.2. Key Aspects

This work in progress aims to propose a tool that reinforces IoT networks’ security by taking advantage of the functions provided by the [RFC8520]. The specific contributions of INXU are listed below:

* Simplify the process of sharing attack signatures that targets or exploits IoT systems;
* Allow a small team of security specialists to protect multiple distinct IoT networks without expose the networks’ privacy;
* Protect the Internet’s ecosystem by hindering distributed attacks that targets its infrastructure.

1.3. INXU Intended Use

The intended use for INXU is in the support of the vulnerability management of diverse heterogeneous IoT networks in scenarios where there is a small team of security specialists (e.g. Smart Cities). It is also intended to be used in scenarios where the end networks need their privacy kept, as Home IoT networks.
The deployment of INXU in networks populated by both IoT and general purpose devices is NOT RECOMMENDED. Due to the greater computing power and wider openness to other attacks, general purpose devices might expose the IoT network to unnecessary risk. Instead of having both types on the same sub-network, we recommend to isolate IoT devices in a separate sub-network as they announce their MUD URLs, and developers should take advantage of MUD’s "controller" and "my-controller" hosts as application gateway between general purpose and IoT devices. In the case of needing a direct communication between the two categories, this could be specified with MUD’s "local-networks" specification.

1.4. Terminology

* INXU: Intra-Network eXposure analyzer Utility.

* INXU Module: a system that crosses data from malicious activities and MUD’s allow-lists to identify and analyze the exposure of vulnerabilities in the connected IoT devices.

* MTD: Malicious Traffic Description data model.

* MTD File: a file that contains descriptions of traffic associated with malicious activities that targets or exploits IoT devices.

* MTD Manager: a system that requests and receives the MTD File. It is responsible for verifying MUD File’s authenticity and integrity. The MTD Manager also requests the MTD File after it’s cache validity expires.

* MTD URL: a URL, configured in the MTD Manager, that locates the MTD File provided by the SOC in charge to protect the client network.

* MTD Server: a web server, managed by the SOC, that hosts the MTD File.

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. The MTD Data Model

The main aim of this data model is to enable describing malicious traffic so that distinct networks can interpret and implement security measures, no matter the connected IoT devices or network topology. Another feature addressed by this data model is allowing the association between the detected exposure and the malicious activity that exploits it, as well as the grouping of vulnerabilities related to the same malicious activity.

The MTD data model makes use of Access Control Lists (ACLs) [RFC8519] under YANG language [RFC7950] to describe the malicious traffic, addressing the classification feature. Furthermore, such as in MUD, there are available two network address abstractions to describe the traffic so that different networks can adapt the description to its context: one abstraction for addresses in the local networks, and the other for using domain names to hosts on the Internet. The data model also includes control fields that support the manageability of the MTD File, so the contained data can be categorized in control data and description data.

This data model covers the complete description of malicious activities from simple attacks with just a few ACE inputs to complex malware that exploits many different vulnerabilities and network resources. Furthermore, to prevent false-positive threat detections, the MTD data model allows inserting context information into the descriptions. The context information in the MTD data model plays the role of specifying the correlation between the described malicious traffic, determining the combinations of exposures that become a risk, and suggesting the action to be taken with each detected threat. This feature supports reducing false-positive detection, as a single traffic exposure may not represent a threat itself.

Basically the context information supports the categorization of a set of vulnerabilities as an effective threat or not. To incorporate the contextual information, we considered the statements listed below, which were assimilated from the IoTSec ontology in [Mozzaquatro2015]:

* A Threat represents an effective risk if the exposure of one or more vulnerabilities can be exploited by an attacker;

* A Vulnerability does not represent a risk by itself, as it can be hidden behind security mechanisms, such as blocking its exposure for potential attackers.
So, in short words, an asset is under threat only when an attacker can exploit one or more vulnerabilities to take advantage of it. Thus, merging the concepts from the ontology and the aims on MTD data model, this document considers the following statements:

* Each Access Control Entry (ACE) has an associated severity defined by the unsigned integer field named risk. When the exposure to the ACE is detected, its risk is considered part of its ACL’s vulnerability classification;

* Each ACL has alert-threshold and risk-threshold fields, both represented by unsigned integer values. When the sum of the exposed ACEs risks reaches the risk threshold, the exposure to the ACL is considered as a vulnerability;

* Each malicious activity described contains a list of critical ACL sets. A malware or attack is classified as a threat when at least one set of critical ACLs contains all its ACLs classified as a vulnerability exposure. When one set’s condition is satisfied, its associated action-to-take has to be triggered.

The three possible actions to be taken are listed below:

* block-all: blocks all ACLs that expose vulnerabilities related to the description. Expected to be used when any traffic associated with the malware or attack threatens the IoT device;

* block-attack: blocks all ACLs that expose vulnerabilities under the attack-traffic group. Expected to be used when only risky ACLs associated with attacks (isolated or in the context of a malware) threatens the IoT device;

* block-not-attack: blocks all ACLs that expose vulnerabilities under the malware’s not-attack-traffic group. Expected to be used when just blocking the operation traffic of a malware prevents exploitation.

2.1. The draft-inxu-mtd YANG Module

A simplified graphical representation of the data models is used in this document. The meaning of the symbols in these diagrams is explained in [RFC8340].
module: draft-inxu-mtd
++--rw mtd-url
++--rw last-update yang:date-and-time
++--rw mtd-signature? inet:uri
++--rw cache-validity? uint8
++--rw malicious-descriptions
++--rw malicious-list* [name]
|  ++--rw name string
|  ++--rw specific-devices* inet:uri
++--rw critical-acl-sets* [name]
|  ++--rw name string
|  ++--rw critical-acl-set* -> /acl:acls/acl/name
|  ++--rw action-to-take draft-inxu-mtd:action-to-take
++--rw to-device-attacks
++--rw traffic-lists
|  ++--rw traffic-list* [name]
|    ++--rw name -> /acl:acls/acl/name
|    ++--rw specific-devices* inet:uri
++--rw from-device-attacks
++--rw traffic-lists
|  ++--rw traffic-list* [name]
|    ++--rw name -> /acl:acls/acl/name
|    ++--rw specific-devices* inet:uri
++--rw to-device-not-attacks
++--rw traffic-lists
|  ++--rw traffic-list* [name]
|    ++--rw name -> /acl:acls/acl/name
|    ++--rw specific-devices* inet:uri
++--rw from-device-not-attacks
++--rw traffic-lists
|  ++--rw traffic-list* [name]
|    ++--rw name -> /acl:acls/acl/name
|   ++--rw specific-devices* inet:uri

++--rw mtd
    ++--rw local-networks? empty
augment /acl:acls/acl:acl:acls/acl:ace:
    ++--rw risk? uint8
augment /acl:acls/acl:acl:
    ++--rw risk-threshold? uint8
    ++--rw alert-threshold? uint8
    ++--rw direction-initiated? mud:direction
2.2. MTD Data Model Definition of Control Fields in the Root "mtd" Container

Here we describe the leaves placed into the "mtd" root container that plays the role of controlling the operation of the MTD File.

2.2.1. mtd-url

Required field that stores the URL where the security authority hosts the MTD File.

2.2.2. mtd-signature

Optional field used to store a URL where the MTD File signature file can be found. It is applicable for offline authenticity verification of the file.

2.2.3. last-update

Required field that contains the timestamp information of the MTD File generation.

2.2.4. cache-validity

Optional field that contains the number of hours to the expiration of the MTD File, starting from "last-update". This field supports integer values between 1 and 160, and if not defined, it is assumed to be 48 hours by the MTD Manager.

2.3. MTD Data Model Definition of Traffic Description Fields in the Root "mtd" Container

The traffic description fields are organized under the "malicious-descriptions" container. The description of a malicious activity allows the aggregation of different attacks, and also other not attack traffic that only turn into malicious when related to the malware operation. This aggregation is important for the security measures decision-making process, as sometimes only a traffic combination makes the threat effective or blocking just one type of traffic can almost disable it, such as the Mirai’s Command and Control traffic.
The description of each leaf is detailed in the Sub-Sections below.

2.3.1. traffic-list

List type field to specify all the traffic in the same direction (incoming/outgoing) that is associated with one attack-traffic or not-attack-traffic.

2.3.1.1. name

Required string field with the name of the ACL that describes one attack-traffic or not-attack-traffic;

2.3.1.2. specific-devices

Optional list to specify the MUD URLs of the IoT devices affected by the described traffic. When this field is filled, INXU only considers the devices here listed as targets of these ACLs.

2.3.2. malicious-descriptions

List that holds the traffic description of all the malicious activities covered by the MTD File.

2.3.2.1. name

Required string field to uniquely name the described malicious activity.

2.3.2.2. specific-devices

Optional list to specify the MUD URLs of the IoT devices that can be affected by the malicious activity. When this field is filled, INXU only considers the devices here listed as affected by this malicious activity.

2.3.2.3. critical-acl-sets

List to specify all the sets of critical ACL and their respective actions to take when all listed ACLs get classified as risky.

2.3.2.3.1. critical-acl-set

List to specify a set of ACLs that, when all listed ACLs get classified as risky, represents a threat caused by the malicious activity.
2.3.2.3.2. action-to-take

Mandatory leaf to specify the action to be taken when the respective set of critical ACLs turns into a threat. The action can be "block-all", "block-attack", or "block-not-attack".

2.3.2.4. to-device-attacks

Container that holds all the malicious activity’s attack traffic targeting an IoT device on the LAN.

2.3.2.5. from-device-attacks

Container that holds all the malicious activity’s attack traffic outgoing from an IoT device on the LAN.

2.3.2.6. not-attack-traffic

Container that holds the traffic not related to attacks, but that turns into malicious when in this context.

2.3.2.6.1. to-device-not-attack-traffic

List with all the ACLs that describe malicious not attack traffic targeting an IoT device on the LAN.

2.3.2.6.2. from-device-not-attack-traffic

List with all the ACLs that describe malicious not attack traffic outgoing from an IoT device on the LAN.

2.4. Augmentation to the ACL Model

This section describes the proposed augments to the ACL model. These augments are responsible for creating the abstraction for the traffic descriptions, enabling the portability of the knowledge to the different networks, and supporting the risk assessment of each vulnerability exposure.

2.4.1. mtd:local-networks

Optional leaf that, when present, means that the current ACE applies to any device on the local IP networks.

2.4.2. direction-initiated

Optional field incorporated from MUD to specify the TCP initiator.
2.4.3. src-dnsname and dst-dnsname

Optional field to enable the usage of DNS domain names to specify the remote host instead of using IPv4 or IPv6 addresses.

2.4.4. risk

Optional unsigned integer field to specify the risk associated with the exposure of the specified ACE. Its default value is 1.

2.4.5. risk-threshold

Optional unsigned integer field to specify the minimal ACL risk value to classify it as a vulnerability exposure. The ACL’s risk value is calculated by the sum of all its child ACE exposures’ risks. Its default value is 1.

2.4.6. alert-threshold

Optional unsigned integer field to specify the minimal ACL risk value to trigger an alert to the exposure. The ACL’s risk value is calculated by the sum of all its child ACE exposures’ risks. Its default value is 1.

2.5. The MTD YANG Model

<CODE BEGINS>file "draft-inxu-mtd@2021-11-22.yang"
module draft-inxu-mtd{
    yang-version 1.1;
    namespace "urn:ietf:params:xml:ns:yang:draft-inxu-mtd";
    prefix draft-inxu-mtd;

    import ietf-yang-types {
        prefix yang;
    }

    import ietf-access-control-list {
        prefix acl;
    }

    import ietf-inet-types {
        prefix inet;
    }

    import ietf-mud {
        prefix mud;
    }
</CODE BEGINS>
import ietf-acldns {
    prefix acldns;
}

organization "IETF IOTOPS (IOT Operations) Working Group";
contact
    "WG Web: http://tools.ietf.org/wg/iotops/
    WG List: iotops@ietf.org
    Author: Sávyo Morais
    savyovm@gmail.com
    Author: Claudio Farias
    cmicelifarias@gmail.com";

description
    "This module is a data-model to describe malicious network traffic.

This module is intended to be serialized via JSON and stored as a file, as described in I-D draft-morais-iotops-inxu-01.

The key words 'MUST', 'MUST NOT', 'REQUIRED', 'SHALL', 'SHALT NOT', 'SHOULD', 'SHOULD NOT', 'RECOMMENDED', 'NOT RECOMMENDED', 'MAY', and 'OPTIONAL' in this document are to be interpreted as described in BCP 14 (RFC 2119) (RFC 8174) when, and only when, they appear in all capitals, as shown here.

Copyright (c) 2022 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 Simplified BSD License set forth in Section 4.c of the IETF Trust’s Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info).

This version of this YANG module is part of I-D draft-morais-iotops-inxu-01; see the I-D itself for full legal notices.";

revision 2022-05-15{
    description
        "Simplifying the data model to one single description of malicious traffic.";
    reference
        "draft-morais-iotops-inxu-01: Intra-Network eXposure analyzer Utility Specification";
}
typedef action-to-take {
    
    type enumeration {
        enum "alert" {
            value 0;
            description
            "Alert user about a risky exposure";
        }
    }
    
    description
    "Type to specify the action to take when a threat is detected";

    enum "block-not-attack" {
        value 1;
        description
        "Block risky exposures of not-attack-traffic and warns users
         about attack-traffic alert exposures";
    }
    enum "block-attack" {
        value 2;
        description
        "Block attack-traffic risky exposures and alert users about
        the block";
    }
    enum "block-all" {
        value 3;
        description
        "Block all risky exposures and alert users about the block";
    }
}

container mtd {
    presence "Enabled for this particular MTD URL";
    description "MTD-related information";
    uses mtd-groupig;
}

grouping mtd-groupig {
    description
    "This grouping is to create a set of definitions of
    malicious traffic of malware and attacks.";

    leaf mtd-url {
        type inet:uri;
        mandatory true;
        description
        "This is the MTD URL associated with the entry found"
in a MTD File";
}

leaf last-update {
  type yang:date-and-time;
  mandatory true;
  description
  "This is intended to be set when the current MTD File is
generated. MTD Managers SHOULD NOT check for updates
between this time plus cache validity."
}

leaf mtd-signature {
  type inet:uri;
  description
  "A URI that resolves to a signature file to verify the
  authenticity of the MTD File."
}

leaf cache-validity {
  type uint8 {
    range "1..168";
  }
  units "hours";
  default "48";
  description
  "The information retrieved from the MTD Server is
  valid for these many hours, after which it should be
  refreshed. MTD Manager implementations do not need to
discard MTD Files beyond this period."
}

container malicious-descriptions {
  description
  "This container has the descriptions of the malware and
attacks that can exploit or target the devices"
  uses malicious-list;
}

container traffic-lists {
  description
  "A grouping for access lists of malicious traffic in the context
  of malware or attacks."
  container traffic-lists {

description
"The access lists of the attack’s malicious traffic
targeting or departing from the local IoT devices."

list traffic-list {
  key "name";
  description
  "Each entry on this list refers to an malicious
  traffic defined by an ACL that should present the
  overall network communication of the attack."

  leaf name {
    type leafref {
      path "/acl:acls/acl:acl/acl:name";
    }
    description
    "The name of the ACL for this entry."
  }

  leaf-list specific-devices {
    type inet:uri;
    description
    "List of MUD URLs of specific devices
    related with the vulnerability";
  }
}
}

grouping malicious-list {
  description
  "A grouping for acess control lists of malicious traffic in the
  context of malware or attacks."

  list malicious-list {
    key "name";
    description
    "The access lists of the malware’s or attack’s malicious
    traffic targeting or departing from the local IoT devices,";

    leaf name {
      type string;
      description
      "The name of the ACL for this entry."
    }
  }
}
"The unique name of the described malicious activity for each entry."
}

leaf-list specific-devices {
  type inet:uri;

  description
    "List of MUD URLs of specific devices related with the vulnerability";
}

list critical-acl-sets{
  key "name";

  description
    "Each list entry represents a malicious activity’s critical set of risky ACL exposures, followed by the action to take when a critical set be detected.";

  leaf name {
    type string;

    description
      "The critical ACL set name";
  }

  leaf-list critical-acl-set {
    type leafref{
      path "/acl:acls/acl:acl/acl:name";
    }

    description
      "A list to specify a set of ACLs that, when all listed ACLs get classified as risky, represents a threat caused by the malicious activity";
  }

  leaf action-to-take {
    type draft-inxu-mtd:action-to-take;
    mandatory true;

    description
      "A leaf to specify the action to be taken when the respective set of critical ACLs turns into a threat.";
  }
}
container to-device-attacks {
    description
    "The set of attack traffic performed by the infected IoT device";

    uses traffic-lists;
}

carrier from-device-attacks {
    description
    "The set of attack traffic performed targeting the infected IoT device";

    uses traffic-lists;
}

carrier to-device-not-attacks {
    description
    "The set of attack traffic performed by the infected IoT device";

    uses traffic-lists;
}

carrier from-device-not-attacks {
    description
    "The set of attack traffic performed targeting the infected IoT device";

    uses traffic-lists;
}
augment "/acl:acls/acl:acl/acl:aces/acl:ace" {  
description  
"Add the risk level information associated to the ACE";
leaf risk {  
type uint8;
default "1";

description  
"Represents risk level of a device being exploited when exposes the device through traffic matching the described ACE.";
}
}

augment "/acl:acls/acl:acl" {  
description  
"Add an acceptable risk threshold and an alert risk threshold to the ACL";
leaf risk-threshold {  
type uint8;
default "1";

description  
"The acceptable risk threshold represents the minimum risk value for the exposure be considered a risk. The actual risk of an ACL is calculated by the sum of all the ACEs that matched on the INXU Module analysis";
}
leaf alert-threshold {  
type uint8;
default "1";

description  
"The acceptable alert threshold represents the minimum risk value for the exposure trigger an alert. The actual risk of an ACL is calculated by the sum of all the ACEs that matched on the INXU Module analysis";
}
}

description  
"Add direction-initiated";
leaf direction-initiated {  
type mud:direction;
description
   "This node matches based on which direction a connection was initiated.";
 }

   description
      "Adding domain names to matching.";
   uses acldns:dns-matches;
 }

   + "/acl:l3/acl:ipv6/acl:ipv6" {
   uses acldns:dns-matches;
   description
      "Adding domain names to matching.";
 }

   deviate not-supported;
   description
      "Field not used in this specification";
 }
 <CODE ENDS>

2.6. MTD File Example

This MTD file describes the traffic of an hipotetic variant of the Mirai botnet. In its attack model, this malware scans for other vulnerable devices in the same network, and its management services (Command and Control, Scan Report, and Loader) are placed in the network edge.

<CODE BEGINS> file "mirai-lan-variant.json"
{
   "draft-inxu-mtd":"mtd",
   "mtd-url":"https://example.com/mirai-lan-variant.json",
   "last-update":"2022-05-15T18:17:00-03:00",
   "malicious-descriptions":{
      "malicious-list":{
         "name":"Mirai-Example",
         "critical-acl-sets":{
            "name":"mirai-prevent-spread",
         }
      }
   }
}<CODE ENDS>
"critical-acl-set": [
    {
        "name":"mirai_infect_v4from",
        "name":"mirai_infect_v4to",
        "name":"mirai_scan_v4from",
        "name":"mirai_scan_v4to"
    },
    "action-to-take": "BLOCK_ATTACK"
],
"name": "mirai-prevet-cnc",
"critical-acl-set": [
    {
        "name": "mirai_cnc_v4from",
        "name": "mirai_cnc_v4to"
    },
    "action-to-take": "BLOCK_N_ATTACK"
],
"name": "mirai-prevet-minimal",
"critical-acl-set": [
    {
        "name": "mirai_cnc_v4from",
        "name": "mirai_cnc_v4to",
        "name": "mirai_infect_v4from",
        "name": "mirai_infect_v4to"
    },
    "action-to-take": "BLOCK_ALL"
],
"to-device-attacks": {
    "traffic-lists": {
        "traffic-list": {
            "name": "mirai_infect_v4to",
            "name": "mirai_scan_v4to"
        }
    }
},
"from-device-attacks": {
    "traffic-lists": {
        "traffic-list": {
            "name": "mirai_infect_v4from",
            "name": "mirai_scan_v4from"
        }
    }
},
"to-device-not-attacks": {
    "traffic-lists": {
"traffic-list": [  
  {"name":"mirai_cnc_v4to"}  
],
"from-device-not-attacks": {  
  "traffic-lists": {  
    "traffic-list": [  
      {"name":"mirai_cnc_v4from"}  
    ]  
  },  
  "from-device-not-attacks": {  
    "traffic-lists": {  
      "traffic-list": [  
        {"name":"mirai_cnc_v4from"}  
      ]  
    }  
  }  
},
"ietf-access-control-list:acls": {  
  "acl": [  
    {  
      "name":"mirai_infect_v4to",  
      "risk-threshold":11,  
      "type": "ipv4-acl-type",  
      "aces": {  
        "ace": [  
          {  
            "name":"infect_23_to",  
            "risk":10,  
            "matches":{  
              "ipv4":{  
                "protocol":6  
              },  
              "tcp":{  
                "destination-port":{  
                  "operator":"eq",  
                  "port":23  
                }  
              }  
            }  
          },  
          {  
            "name":"infect_2323_to",  
            "risk":10,  
            "matches":{  
              "ipv4":{  
                "protocol":6  
              }  
            }  
          }  
        ]  
      }  
    }  
  ]
"name":"bin_download_to",
"risk":1,
"matches":{
   "ipv4":{
      "protocol":6
   },
   "tcp":{
      "source-port":{
         "operator":"eq",
         "port":80
      }
   }
}
},

"name":"mirai_scan_v4to",
"risk-threshold":11,
"type": "ipv4-acl-type",
"aces": {
   "ace": [ {
      "name":"scan_23_to",
      "risk":10,
      "matches":{
         "ietf-mud:mud":{
            "local-networks":[ null ]
         },
         "ipv4":{
            "protocol":6
         },
         "tcp":{
            "source-port":{
               "operator":"eq",
               "port":80
            }
         }
      }
   }
   ]
}
}
"port":23
},
}
],
"name":"scan_2323_to",
"risk":10,
"matches":{
  "ietf-mud:mud":{
    "local-networks": [null]
  },
  "ipv4":{
    "protocol": 6
  },
  "tcp":{
    "source-port":{
      "operator": "eq",
      "port": 2323
    }
  }
}
},
"name": "scan_report_to",
"risk": 1,
"matches":{
  "ipv4":{
    "ietf-acldns:dst-dnsname": "urn:ietf:params:mud:gateway",
    "protocol": 6
  },
  "tcp":{
    "source-port":{
      "operator": "eq",
      "port": 48101
    }
  }
}
],
"name": "mirai_infect_v4from",
"risk-threshold": 11,
"type": "ipv4-acl-type",
"aces": {
  

"ace": [
{
"name": "infect_23_from",
"risk": 10,
"matches": {
"ipv4": {
"ietf-acldns:dst-dnsname": "urn:ietf:params:mud:gateway",
"protocol": 6
},
"tcp": {
"source-port": {
"operator": "eq",
"port": 23
}
}
},
{
"name": "infect_2323_from",
"risk": 10,
"matches": {
"ipv4": {
"ietf-acldns:dst-dnsname": "urn:ietf:params:mud:gateway",
"protocol": 6
},
"tcp": {
"source-port": {
"operator": "eq",
"port": 2323
}
}
}
},
{
"name": "bin_download_from",
"risk": 1,
"matches": {
"ipv4": {
"ietf-acldns:dst-dnsname": "urn:ietf:params:mud:gateway",
"protocol": 6
},
"tcp": {
"destination-port": {
"operator": "eq",
"port": 80
}
}
}
]


```json
{
    "name": "mirai_scan_v4from",
    "risk-threshold": 11,
    "type": "ipv4-acl-type",
    "aces": [
        {
            "name": "scan_23_from",
            "risk": 10,
            "matches": {
                "ietf-mud:mud": {
                    "local-networks": [ null ]
                },
                "ipv4": {
                    "protocol": 6
                },
                "tcp": {
                    "destination-port": {
                        "operator": "eq",
                        "port": 23
                    }
                }
            }
        },
        {
            "name": "scan_2323_from",
            "risk": 10,
            "matches": {
                "ietf-mud:mud": {
                    "local-networks": [ null ]
                },
                "ipv4": {
                    "protocol": 6
                },
                "tcp": {
                    "destination-port": {
                        "operator": "eq",
                        "port": 2323
                    }
                }
            }
        }
    ]
}
```

Morais & Farias  Expires 27 November 2022  [Page 27]


```
}

{
  "name": "scan_report_from",
  "risk": 1,
  "matches": {
    "ipv4": {
      "ietf-acldns:dst-dnsname": "urn:ietf:params:mud:gateway",
      "protocol": 6
    },
    "tcp": {
      "destination-port": {
        "operator": "eq",
        "port": 48101
      }
    }
  }
}

{
  "name": "mirai_cnc_v4to",
  "type": "ipv4-acl-type",
  "aces": {
    "ace": [
      {
        "name": "cnc_socket_to",
        "risk": 1,
        "matches": {
          "ipv4": {
            "ietf-acldns:dst-dnsname": "urn:ietf:params:mud:gateway",
            "protocol": 6
          },
          "tcp": {
            "source-port": {
              "operator": "eq",
              "port": 2030
            }
          }
        }
      }
    ]
  }
}

{
  "name": "mirai_cnc_v4from",
  ```
3. The Intra-Network eXposure analyzer Utility

INXU was designed to have as main features: (i) enable quick responses to new vulnerabilities; (ii) allow mitigation of the damages of a new vulnerability, simultaneously in multiple distinct networks; and (iii) enable a decision-making process about security measures on the network edge, avoiding the disclosure of private information to third parties.

To cover these requirements, INXU enables a security expert team - such as a SOC or an ISP Abuse Desk - to describe the traffic of ongoing malicious activities using the MTD data model (more details of the MTD data model are discussed in Section 3). With these descriptions, the security experts team can use the INXU to prevent multiple distinct networks when releasing new MTD Files for every new malicious activity discovered, in a process similar to the antivirus programs vaccines.
3.1. INXU Architecture and Components

The ASCII diagram below shows the architecture of INXU. The proposed architecture contains 4 main components: MTD Server, MTD Manager, INXU Module, and MUD manager [RFC8520].

The MTD Server is responsible for storing and delivering the malicious traffic descriptions made by a security expert. This component was designed to enable trusted third-party specialists to share knowledge about well-known malicious activities affecting IoT and allow IoT networks to make use of this knowledge to protect themselves. The MTD Server is composed by a HTTPS server that hosts and delivers the MTD File for the clients. The MTD File is a file where the network traffic associated with malicious activities are described to INXU. This component also contains data for version control, authenticity, and validity time. The content of a MTD File is defined by a security expert in a JSON file, following the YANG data model described in the Section 2.
The MTD Manager has the function of managing the MTD File on the system. It is responsible for requesting the file to the MTD Server, verifying authenticity, and requesting a new file when the current file validity expires. The default way to ensure MTD File authenticity is by HTTPS protocol, but the MTD Manager can also use the means described in the Section 3.1 of the [RFC2818]. At the end of the process, the MTD Manager forwards the MTD File to the INXU Module.

The INXU (Intra-Network eXposure analyzer Utility) module is the main component of this proposal. It is responsible for verifying all the network communications trying to identify possible exposures to malicious traffic. To do this, the INXU Module compares the malicious traffic described in a MTD File with the network graph generated by MUD manager. The exposure analysis process is detailed in Section 3.5.

3.2. Workflow

The workflow adopted to INXU may vary, but it will mostly follow the process described below.

1. MTD Manager fetches the MTD File.
2. After confirming MTD File authenticity, the MTD Manager sends it for the INXU Module.
3. The MUD manager sends the network communication graph -- including the devices’ MUD URLs -- to the INXU Module.
4. INXU Module identifies potential vulnerability exposures into the network.
5. INXU analyzes the detected vulnerabilities to evaluate if they represent an effective threat. After detecting threats, INXU may act as an Intrusion Prevention System and block the exposures, or serve as input source for other security systems, depending on the implementation.
6. If the MUD manager detects any change in the network topology, or the MTD Manager gets new definitions from MTD Files, the process returns to step 4.
3.3. Acquiring a MTD File

The main method for acquiring a MTD File is by configuring the MTD URL into the MTD Manager. The MTD URL is a Universal Resource Locator (URL) [RFC3986] provided by the Security Experts team designated for protecting the network.

MTD URLs MUST use the "https" scheme [RFC7230].

An alternative manner for acquiring a MTD File is by manually importing and its respective signature file into the MTD Manager. The mechanisms for doing so are not described in this document.

3.4. Processing a MTD URL

Disclaimer: The specification in this section is in our roadmap but still not done. Our initial intention is to use the same specification as [RFC8520] in Section 1.6. To simplify understanding, we copied the original MUD text, pasted it below, and replaced the MUD references with MTD.

MTD Managers that are able to do so SHOULD retrieve MTD URLs and signature files as per [RFC7230], using the GET method [RFC7231]. They MUST validate the certificate using the rules in [RFC2818], Section 3.1.

Requests for MTD URLs SHOULD include an "Accept" header field ([RFC7231], Section 5.3.2) containing "application/mtd+json", an "Accept-Language" header field ([RFC7231], Section 5.3.5), and a "User-Agent" header field ([RFC7231], Section 5.5.3).

MTD Managers SHOULD automatically process 3xx response status codes.

If a MTD Manager is not able to fetch a MTD URL, other means MAY be used to import the MTD File and its associated signature file. So long as the signature of the file can be validated, the file can be used. In such environments, controllers SHOULD warn administrators when cache-validity expiry is approaching so that they may check for new files.

3.5. INXU Vulnerability Analysis Process

The exposure analysis algorithm of the INXU Module uses malicious traffic descriptions from a MTD File to compare with the IoT traffic flows allowed by MUD -- provided by the network communication graph generated by MUD manager -- and tries to detect vulnerabilities on the network. In this context, INXU identifies one exposure when some graph edge matches with any entry of the MTD File.
Based on the MUD files, each host expected to communicate with the IoT devices, or the IoT devices themselves, are represented by nodes on the network communication graph generated by MUD manager. The host network address represents the nodes, and in the case of IoT devices on the LAN, the MUD URL is associated with the node information. The graph edges represent TCP or UDP sockets, or ICMP communications, where a directed edge represents a communication path.

3.5.1. Exposure Identification

For each IoT node in the MUD-based communication graph, the Exposure Identification process verifies if five information match between edge and ACEs: source and destination host addresses, communication protocol, and source and destination ports -- for transport protocols -- or ICMP message type and code. We only consider an exposure when all five information match.

The ACEs considered here MUST be applicable for any device OR include the IoT device’s MUD URL in the specific-devices list.

A match on source or destination host address happens when the addresses are equals OR when the ACE uses the local address abstraction and the node is local.

Protocols match when the specified protocols (TCP, UDP, ICMP, or any) are equal both on ACE and edge OR when the ACE specifies any protocol.

For the ICMP message type and code or for transport’s source and destination ports, a match happens when the specified values are equals OR when the ACE specifies any value.

3.5.2. ACL Risk Assessment

Each vulnerability exposure is associated with an ACE and is in the context of an ACL. Therefore, a set of vulnerability exposures of a device becomes a risk when the sum of their ACE risks is bigger than the ACL’s risk threshold. There is also a possibility of triggering an alert state when the ACL’s risk exceeds the alert threshold.

The risk threshold SHOULD be equals or bigger than the alert threshold.
3.5.3. Threat Analysis

After assessing the risk of each ACL, the next step in the process is the threat analysis. This analysis iterates over the list of the critical ACL sets of a malicious activity.

In this step, the INXU Module verifies if all the ACLs contained in a critical set are classified as risky for a device. If this condition becomes true, the INXU Module SHOULD take the action specified in the set’s action-to-take field. If a malicious activity threatens the device with more than one set of critical ACLs, the INXU Module MUST take an action based into a merge of all the threatening sets’ action-to-take.

4. Further Considerations and Next Steps

During the development of INXU, we found some important points that could further enhance the proposal in the near future. First of all, although INXU sticks to the Network and Transport layers, many recently reported DDoS attacks exploited the DNS platform to cause damage. This issue requires some treatment in this application layer protocol of the TCP/IP model. As it is a crucial application for the Internet’s functioning as we know it today, it is impossible to block traffic over the protocol completely, but we believe that some level of filtering will not negatively impact the devices’ usability nor the network’s performance.

Another interesting future direction is that although INXU allows identifying, classifying, and mitigating malicious activities on the other hand it does that without any intervention from the user. All the blocking processes do not allow the end-users intervention on the blocks and may lead them to not adopt INXU. An option to overcome this issue is by integrating Software Bill of Materials (SBOM) related information into the MTD data model and in the Threat Analysis process, and allowing end-users feedback on blocking decisions. This may reduce INXU’s impact on usability with low security loss and consequently improve its adoption.

Also in this sense, we could use the MTD as a standard data model for attacks signatures involving IoT. It is a useful way to share how attacks can alter the network traffic to be used in controlled experiments and simulations. Also it can be seen also as a systematic way to share information on attacks -- in this sense network administrators, scientists and security analysts could have the same view over a given event in the network.
Finally, also coming from the previous statement, INXU’s output could be used as an input filter for IPS/IDS systems in order to prevent attacks and any other malicious event in the network. Since by using the MTD we could classify the traffic into appropriate or not. Furthermore INXU -- specially the MTD -- could be paired with an AI engine to learn about new network patterns and classify them as an attack or some new device in the network -- the system could write some new MTDs as it learns from the network.

5. Security Considerations

Since INXU uses MUD as a data source, the problems presented at the Security Considerations session of the [RFC8520] are still valid for this proposal, and some new ones arise.

The first new risk is the possibility of INXU causing Denial of Service on their protected IoT devices depending on how the malicious activities are described in the MTD File. To prevent this issue, while describing a malicious activity, the Security Specialist SHOULD be as specific as possible by describing, for example, the specific devices that can be affected by the attack or malware and being assertive while defining ACE risks and ACL risk thresholds.

As with MUD, the MTD Manager may receive a fake MTD File from a rogue MTD Server with a certificate issued by an accredited certification authority (CA). In this case, the same MUD mitigations apply: First, if the signer changes, this may be flagged as an exception by the MTD manager. Second, if the MTD file also changes, the MTD manager SHOULD seek administrator approval (it should do this in any case). In all circumstances, the MUD manager MUST maintain a cache of trusted CAs for this purpose. When such a rogue is discovered, it SHOULD be removed.

Finally, INXU is not effective against attacks that are occurring prior to a new MTD file arriving or ongoing at the moment of an update. The classification of the attack is not accurate since it does not know the rules. A countermeasure is to use an anomaly detection system to identify such attacks. INXU is not responsible for that part.

Further security considerations might arise during this document’s evolution.

6. IANA Considerations

This memo includes no request to IANA.

7. References
7.1. Normative References


7.2. Informative References


Authors’ Addresses

Sávyo Vinícius de Morais
IFRN
Natal-
Brazil
Email: savvyvm@gmail.com

Cláudio Miceli de Farias
UFRJ
Rio de Janeiro-
Brazil
Email: cmicelifarias@gmail.com
Export of Forwarding Path Delay in IPFIX
draft-tgraf-opsawg-ipfix-inband-telemetry-00

Abstract

This document introduces new IP Flow Information Export (IPFIX) information elements to expose the Inband Telemetry measured forwarding path delay in passport and postcard mode on the transit and decapsulation nodes.

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 9 January 2023.

Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.
Table of Contents

1. Introduction ............................................. 3
2. IP One-Way Delay Hybrid Type I Passive Registry Entries ... 4
   2.1. Summary ........................................... 4
   2.1.1. ID (Identifier) ............................... 4
   2.1.2. Name .......................................... 4
   2.1.3. Name .......................................... 4
   2.1.4. URI ........................................... 4
   2.2. Description ...................................... 5
   2.3. Change Controller ................................. 5
   2.4. Version of Registry Format ......................... 5
3. Metric Definition ......................................... 5
   3.1. Reference Definition .............................. 5
   3.2. Fixed Parameters ................................. 6
4. Method of Measurement ..................................... 6
   4.1. Reference Methods ................................. 6
   4.2. Packet Stream Generation .......................... 6
   4.3. Traffic Filtering (Observation) Details .......... 6
   4.4. Sampling Distribution ............................. 6
   4.5. Runtime Parameters and Data Format ............... 7
   4.6. Roles ........................................... 7
5. Output .................................................. 8
   5.1. Type ............................................. 8
   5.2. Reference Definition .............................. 8
      5.2.1. Mean ....................................... 8
      5.2.2. Min ....................................... 8
      5.2.3. Max ....................................... 9
      5.2.4. Sum ....................................... 9
      5.2.5. Metric Units ............................... 10
      5.2.6. Calibration ................................. 10
   5.3. Administrative Items ............................... 10
      5.3.1. Status .................................... 10
      5.3.2. Requester ................................. 10
      5.3.3. Revision ................................. 10
      5.3.4. Revision Date ............................. 11
   5.4. Comments and Remarks .............................. 11
6. IPFIX Information Elements ............................... 11
7. Use Cases .............................................. 11
1. Introduction

With Inband Telemetry, defined in In-situ OAM
[I-D.ietf-ippm-ioam-deployment], Path Tracing
[I-D.filsfils-spring-path-tracing] and In-situ Flow Information
Telemetry [I-D.song-opsawg-ifit-framework], the path delay between
two endpoints is measured by inserting a timestamp in the packet.

Inband Telemetry can be distinguished between two modes. Passport
mode where only the last hop in the forwarding path of the Inband
Telemetry domain exposes all the metrics and postcard mode where the
transit nodes also expose metrics. In both modes the forwarding path
is exposed thus allowing to determine how much delay has been
accumulated hop by hop.

This document defines eight new IPFIX Information Elements (IEs) with
their four corresponding entries in the performance metrics registry
to expose the forwarding path delay on the transit and decapsulation
nodes.

The delay is measured by calculating the difference between the
timestamp imposed with Inband Telemetry in the packet at the
encapsulation node and the timestamp exported in the IPFIX flow
record from the transit and decapsulation nodes. Depending on the
IE, the lowest, highest or the sum of measured delay is being
exported.
2. IP One-Way Delay Hybrid Type I Passive Registry Entries

This section specifies four Registry Entries for the Hybrid Type I Passive assessment of IP One-Way Delay.

All column entries besides the ID, Name, Description, and Output Reference Method categories are the same; thus, this section defines four closely related Registry Entries. As a result, IANA has assigned corresponding URLs to each of the four Named Metrics.

2.1. Summary

This category includes multiple indexes to the Registry Entry: the element ID and Metric Name.

2.1.1. ID (Identifier)

<insert a numeric Identifier, an integer, TBD>

2.1.2. Name

IANA has allocated the numeric Identifiers TBD1-4 for the four Named Metric Entries in this section

2.1.3. Name

TBD1: OWDelay_HybridType1_Passive_IP_RFCTBD_Seconds_Mean
TBD2: OWDelay_HybridType1_Passive_IP_RFCTBD_Seconds_Min
TBD3: OWDelay_HybridType1_Passive_IP_RFCTBD_Seconds_Max
TBD4: OWDelay_HybridType1_Passive_IP_RFCTBD_Seconds_Sum

2.1.4. URI

URL: https://www.iana.org/assignments/performance-metrics/OWDelay_HybridType1_Passive_IP_RFCTBD_Seconds_Mean
URL: https://www.iana.org/assignments/performance-metrics/OWDelay_HybridType1_Passive_IP_RFCTBD_Seconds_Min
URL: https://www.iana.org/assignments/performance-metrics/OWDelay_HybridType1_Passive_IP_RFCTBD_Seconds_Max
URL: https://www.iana.org/assignments/performance-metrics/OWDelay_HybridType1_Passive_IP_RFCTBD_Seconds_Sum
2.2. Description

This metric assesses the one-way delay of IP packets constituting a single connection, exchanged between two hosts. We consider the measurement of one-way delay based on a single Observation Point (OP) [RFC7011] somewhere in the network. The output is the one-way delay for all successfully exchanged packets expressed as the <statistic> of their conditional delay distribution, where <statistic> is one of:

* Mean
* Min
* Max
* Sum

2.3. Change Controller

IETF

2.4. Version of Registry Format

1.0

3. Metric Definition

This category includes columns to prompt the entry of all necessary details related to the metric definition, including the immutable document reference and values of input factors, called "Fixed Parameters".

3.1. Reference Definition


Section 3.4 of [RFC7679] provides the reference definition of the singleton (single value) one-way delay metric. Section 4.4 of [RFC7679] provides the reference definition expanded to cover a multi-value sample. Note that terms such as "singleton" and "sample" are defined in section 2 of [RFC2330].
With the OP [RFC7011] typically located between the hosts participating in the IP connection, the one-way delay metric requires one individual measurement between the OP and sourcing host, such that the Spatial Composition [RFC6049] of the measurements yields a one-way delay singleton.

3.2. Fixed Parameters

Traffic Filters:

IPv4 header values:
  DSCP: Set to 0

IPv6 header values:
  DSCP: Set to 0
  Hop Count: Set to 255
  Flow Label: Set to 0
  Extension Headers: None

4. Method of Measurement

This category includes columns for references to relevant sections of the RFC(s) and any supplemental information needed to ensure an unambiguous method for implementations.

4.1. Reference Methods

The foundational methodology for this metric is defined in section 4 of [RFC7323] using the Timestamps option with modifications that allow application at a mid-path OP [RFC7011].

The Traffic Filter at the OP is configured to observe a single IP connection.

4.2. Packet Stream Generation

N/A

4.3. Traffic Filtering (Observation) Details

The Fixed Parameters above give a portion of the Traffic Filter. Other aspects will be supplied as Runtime Parameters (below).

4.4. Sampling Distribution

This metric requires a partial sample of all packets that qualify according to the Traffic Filter criteria.
4.5. Runtime Parameters and Data Format

Runtime Parameters are input factors that must be determined, configured into the measurement system, and reported with the results for the context to be complete.

Src: The IP address of the host in the host A Role (format ipv4-address-no-zone value for IPv4 or ipv6-address-no-zone value for IPv6; see section 4 of [RFC6991]).

Dst: The IP address of the host in the host B Role (format ipv4-address-no-zone value for IPv4 or ipv6-address-no-zone value for IPv6; see section 4 of [RFC6991]).

TTL or Hop Limit: Set at desired value.

DSCP: Set at desired value.

IPv6 Flow Label: Set at desired value.

Timestamp: The timestamp when the packet is being received at IOAM encapsulation node. Format depends on Inband Telemetry implementation. For IOAM, Section 4.4.1 of [RFC9197] describes what kind of timestamps are supported. Section 4.4.2.3 and 4.4.2.4 describe where the timestamp is being inserted. For Path Tracing, Section 4.1 of [I-D.filsfils-spring-path-tracing] describes what kind of timestamps are supported. Section 9.2 describe the SRH path tracing TLV where the timestamp is being inserted.

4.6. Roles

host A: Launches the IP packet to open the connection. The Role of "host A" is synonymous with the IP address used at host A.

host B: Receives the IP packet to open the connection. The Role of "host B" is synonymous with the IP address used at host B.

Encapsulation Node: Receives the IP packet to open the connection and encapsulates the timestamp into the packet. The Role of "Encapsulation Node" is synonymous with the timestamp inserted in the packet.

Transit Node: Receives the IP packet to open the connection and measures the delay between the timestamp in the packet and the timestamp when the packet was received.

Decapsulation Node: Receives the IP packet to open the connection
and measures the delay between the timestamp in the packet and the
timestamp when the packet was received and removes the IOAM header
from the packet.

5. Output

This category specifies all details of the output of measurements
using the metric.

5.1. Type

OWDelay Types are discussed in the subsections below.

5.2. Reference Definition

For all output types:

OWDelay_HybridType1_Passive_IP: The one-trip delay of one IP packet
is a Singleton

For each <statistic>, Singleton one of the following subsections
applies.

5.2.1. Mean

The mean SHALL be calculated using the conditional distribution of
all packets with a finite value of one-way delay (undefined delays
are excluded) -- a single value, as follows:

See section 4.1 of [RFC3393] for details on the conditional
distribution to exclude undefined values of delay, and see section 5
of [RFC6703] for background on this analysis choice.

See section 4.2.2 of [RFC6049] for details on calculating this
statistic; see also section 4.2.3 of [RFC6049].

Mean: The time value of the result is expressed in units of seconds,
as a positive value of type decimal64 with fraction digits = 9
(see section 9.3 of [RFC6020]) with a resolution of
0.000000001 seconds (1.0 ns), and with lossless conversion to/from
the 64-bit NTP timestamp as per section 6 of [RFC5905].

5.2.2. Min

The minimum SHALL be calculated using the conditional distribution of
all packets with a finite value of one-way delay (undefined delays
are excluded) -- a single value, as follows:
See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and see section 5 of [RFC6703] for background on this analysis choice.

See section 4.3.2 of [RFC6049] for details on calculating this statistic; see also section 4.3.3 of [RFC6049].

Min: The time value of the result is expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with a resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of [RFC5905].

5.2.3. Max

The maximum SHALL be calculated using the conditional distribution of all packets with a finite value of one-way delay (undefined delays are excluded) -- a single value, as follows:

See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and see section 5 of [RFC6703] for background on this analysis choice.

See section 4.3.2 of [RFC6049] for a closely related method for calculating this statistic; see also section 4.3.3 of [RFC6049]. The formula is as follows:

\[
\text{Max} = \{\text{FiniteDelay}[j]\}
\]

such that for some index, \( j \), where \( 1 \leq j \leq N \)

\[
\text{FiniteDelay}[j] \geq \text{FiniteDelay}[n] \text{ for all } n
\]

Max: The time value of the result is expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with a resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of [RFC5905].

5.2.4. Sum

The sum SHALL be calculated using the conditional distribution of all packets with a finite value of one-way delay (undefined delays are excluded) -- a single value, as follows:

See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and see section 5 of [RFC6703] for background on this analysis choice.
See section 4.3.5 of [RFC6049] for details on calculating this statistic. However in this case FiniteDelay or MaxDelay MAY be used.

**Sum**: The time value of the result is expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with a resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of [RFC5905].

5.2.5. Metric Units

The `<statistic>` of one-way delay is expressed in seconds, where `<statistic>` is one of:

* Mean
* Min
* Max
* Sum

The one-way delay of the IP connection singleton is expressed in seconds.

5.2.6. Calibration

Passive Measurements at an OP could be calibrated against an Active Measurement at host A where the Active Measurement represents the ground truth.

5.3. Administrative Items

5.3.1. Status

Current

5.3.2. Requester

This RFC

5.3.3. Revision

1.0
5.3.4. Revision Date

RFC Date

5.4. Comments and Remarks

None

6. IPFIX Information Elements

This section defines and describes the new IPFIX IEs.

PathDelayMeanDeltaMicroseconds
16-bit unsigned integer that identifies the mean path delay.

PathDelayMeanDeltaNanoseconds
32-bit unsigned integer that identifies the mean path delay.

PathDelayMinDeltaMicroseconds
16-bit unsigned integer that identifies the lowest path delay.

PathDelayMinDeltaNanoseconds
32-bit unsigned integer that identifies the lowest path delay.

PathDelayMaxDeltaMicroseconds
16-bit unsigned integer that identifies the highest path delay.

PathDelayMaxDeltaNanoseconds
32-bit unsigned integer that identifies the highest path delay.

PathDelaySumDeltaMicroseconds
32-bit unsigned integer that identifies the sum of the path delay.

PathDelaySumDeltaNanoseconds
64-bit unsigned integer that identifies the sum of the path delay.

7. Use Cases

The measured forwarding path delay can be aggregated with Flow Aggregation as defined in [RFC7015] to the following device and control-plane dimensions to determine:

* With node id and egressInterface(IE14), on which node which logical egress interfaces have been contributing to how much delay.
* With node id and egressPhysicalInterface(253), on which node which physical egress interfaces have been contributing to how much delay.

* With ipNextHopIPv4Address(IE15) or ipNextHopIPv6Address(IE62), the forwarding path to which next-hop IP contributed to how much delay.

* With mplsTopLabelIPv4Address(IE47) or srhActiveSegmentIPv6 from [I-D.tgraf-opsawg-ipfix-srv6-srh], the forwarding path to which MPLS top label IPv4 address or SRv6 active segment contributed to how much delay.

* BGP communities are often used for setting a path priority or service selection. With bgpDestinationExtendedCommunityList(488) or bgpDestinationCommunityList(485) or bgpDestinationLargeCommunityList(491) which group of prefixes accumulated at which node how much delay.

* With destinationIPv4Address(13), destinationTransportPort(11), protocolIdentifier (4) and sourceIPv4Address(IE8), the forwarding path delay on each node from each IPv4 source address to a specific application in the network.

8. IANA Considerations

This document requests IANA to create new IE(s) (see table 1) and assign the following initial code points.
<table>
<thead>
<tr>
<th>Element ID</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>TBD5</td>
<td>PathDelayMeanDeltaMicroseconds</td>
</tr>
<tr>
<td>TBD6</td>
<td>PathDelayMeanDeltaNanoseconds</td>
</tr>
<tr>
<td>TBD7</td>
<td>PathDelayMinDeltaMicroseconds</td>
</tr>
<tr>
<td>TBD8</td>
<td>PathDelayMinDeltaNanoseconds</td>
</tr>
<tr>
<td>TBD9</td>
<td>PathDelayMaxDeltaMicroseconds</td>
</tr>
<tr>
<td>TBD10</td>
<td>PathDelayMaxDeltaNanoseconds</td>
</tr>
<tr>
<td>TBD11</td>
<td>PathDelaySumDeltaMicroseconds</td>
</tr>
<tr>
<td>TBD12</td>
<td>PathDelaySumDeltaNanoseconds</td>
</tr>
</tbody>
</table>

Table 1: Creates IEs in the "IPFIX Information Elements" registry

Note to the RFC-Editor:

* Please replace TBD5 - TBD12 with the values allocated by IANA
* Please replace the [RFC-to-be] with the RFC number assigned to this document

8.1. PathDelayMeanDeltaMicroseconds

Name: PathDelayMeanDeltaMicroseconds  ElementID: TBD5  Description:
This Information Element identifies the mean path delay in microseconds.  Abstract Data Type: unsigned16  Data Type Semantics: OctedDelta  Reference: [RFC-to-be], xxx
8.2. PathDelayMeanDeltaNanoseconds

Name: PathDelayMeanDeltaNanoseconds ElementID: TBD6 Description: This Information Element identifies the mean path delay in nanoseconds. Abstract Data Type: unsigned32 Data Type Semantics: OctedDelta Reference: [RFC-to-be], xxx

8.3. PathDelayMinDeltaMicroseconds

Name: PathDelayMinDeltaMicroseconds ElementID: TBD7 Description: This Information Element identifies the lowest path delay in microseconds. Abstract Data Type: unsigned16 Data Type Semantics: OctedDelta Reference: [RFC-to-be], xxx

8.4. PathDelayMinDeltaNanoseconds

Name: PathDelayMinDeltaNanoseconds ElementID: TBD8 Description: This Information Element identifies the lowest path delay in nanoseconds. Abstract Data Type: unsigned32 Data Type Semantics: OctedDelta Reference: [RFC-to-be], xxx

8.5. PathDelayMaxDeltaMicroseconds

Name: PathDelayMaxDeltaMicroseconds ElementID: TBD9 Description: This Information Element identifies the highest path delay in microseconds. Abstract Data Type: unsigned16 Data Type Semantics: OctedDelta Reference: [RFC-to-be], xxx

8.6. PathDelayMaxDeltaNanoseconds

Name: PathDelayMaxDeltaNanoseconds ElementID: TBD10 Description: This Information Element identifies the highest path delay in nanoseconds. Abstract Data Type: unsigned32 Data Type Semantics: OctedDelta Reference: [RFC-to-be], xxx

8.7. PathDelaySumDeltaMicroseconds

Name: PathDelaySumDeltaMicroseconds ElementID: TBD11 Description: This Information Element identifies the sum of the path delay in microseconds. Abstract Data Type: unsigned32 Data Type Semantics: OctedDelta Reference: [RFC-to-be], xxx

8.8. PathDelaySumDeltaNanoseconds

Name: PathDelaySumDeltaNanoseconds ElementID: TBD12 Description: This Information Element identifies the sum of the path delay in nanoseconds. Abstract Data Type: unsigned64 Data Type Semantics: OctedDelta Reference: [RFC-to-be], xxx
9. Operational Considerations

9.1. Time Accuracy

The same recommendation as defined in section 4.5 of [RFC5153] for IPFIX applies in terms of clock precision to this document as well.

9.2. Mean Delay

The mean (average) path delay can be calculated by dividing the PathDelaySumDeltaMicroseconds(TBD5) or PathDelaySumDeltaNanoseconds(TBD6) by the packetDeltaCount(2) at the IPFIX data collection.

9.3. IOAM Packet Time Stamps

For IOAM, Section 4.4.1 of [RFC9197] describes what kind of timestamps are supported. Section 4.4.2.3 and 4.4.2.4 describe where the timestamp is being inserted.

For Path Tracing, Section 4.1 of [I-D.filsfils-spring-path-tracing] describes what kind of timestamps are supported. Section 9.2 describe the SRH path tracing TLV where the timestamp is being inserted.

10. Security Considerations

There are no significant extra security considerations regarding the allocation of these new IPFIX IEs compared to [RFC7012].

11. Acknowledgements

The authors would like to thank xxx for their review and valuable comments.

12. References

12.1. Normative References


12.2. Informative References
[I-D.filsfils-spring-path-tracing]
Filsfils, C., Abdelsalam, A., Garvia, P. C., Yufit, M.,
Graf, T., Su, Y., Matsushima, S., and M. Valentine, "Path
Tracing in SRv6 networks", Work in Progress, Internet-
Draft, draft-filsfils-spring-path-tracing-01, 30 May 2022,
<https://www.ietf.org/archive/id/draft-filsfils-spring-
path-tracing-01.txt>.

[I-D.ietf-ippm-ioam-deployment]
Brockners, F., Bhandari, S., Bernier, D., and T. Mizrahi,
"In-situ OAM Deployment", Work in Progress, Internet-
Draft, draft-ietf-ippm-ioam-deployment-01, 11 April 2022,
<https://www.ietf.org/archive/id/draft-ietf-ippm-ioam-
deployment-01.txt>.

[I-D.song-opsawg-ifit-framework]
Song, H., Qin, F., Chen, H., Jin, J., and J. Shin, "A
Framework for In-situ Flow Information Telemetry", Work in
Progress, Internet-Draft, draft-song-opsawg-ifit-
framework-17, 22 February 2022,
<https://www.ietf.org/archive/id/draft-song-opsawg-ifit-
framework-17.txt>.

[I-D.tgraf-opsawg-ipfix-srv6-srh]
Graf, T., Claise, B., and P. Francois, "Export of Segment
Routing IPv6 Information in IP Flow Information Export
(IPFIX)", Work in Progress, Internet-Draft, draft-tgraf-
opsawg-ipfix-srv6-srh-04, 28 June 2022,
<https://www.ietf.org/archive/id/draft-tgraf-opsawg-ipfix-
srv6-srh-04.txt>.

[RFC2330] Paxson, V., Almes, G., Mahdavi, J., and M. Mathis,
"Framework for IP Performance Metrics", RFC 2330,
DOI 10.17487/RFC2330, May 1998,

[RFC3393] Demichelis, C. and P. Chimento, "IP Packet Delay Variation
Metric for IP Performance Metrics (IPPM)", RFC 3393,
DOI 10.17487/RFC3393, November 2002,

[RFC5153] Boschi, E., Mark, L., Quittek, J., Stiemerling, M., and P.
Aitken, "IP Flow Information Export (IPFIX) Implementation
Guidelines", RFC 5153, DOI 10.17487/RFC5153, April 2008,


Authors’ Addresses

Thomas Graf
Swisscom
Binzring 17
CH-8045 Zurich
Switzerland
Email: thomas.graf@swisscom.com

Benoit Claise
Huawei
Email: benoit.claise@huawei.com

Alex Huang Feng
INSA-Lyon
Lyon
France
Email: alex.huang-feng@insa-lyon.fr
Export of Segment Routing IPv6 Information in
IP Flow Information Export (IPFIX)
draft-tgraf-opsawg-ipfix-srv6-srh-05

Abstract

This document introduces new IP Flow Information Export (IPFIX) information elements to identify the SRv6 Segment Routing Header dimensions, the SRv6 Control Plane Protocol and the SRv6 Endpoint Behavior that traffic is being forwarded with.

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 25 January 2023.

Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.
1. Introduction

A new type of Routing Extension Header called Segment Routing Header (SRH) is defined by [RFC8754] which is used for applying Segment Routing (SR) on the IPv6 data plane.
Three routing protocol extensions, OSPFv3 Extensions [I-D.li-lsr-ospfv3-srv6-extensions], IS-IS Extensions [I-D.ietf-lsr-isis-srv6-extensions], BGP Prefix Segment Identifiers (Prefix-SIDs) [I-D.ietf-bess-srv6-services] and one Path Computation Element Communication Protocol (PCEP) Extension [I-D.ietf-pce-segment-routing-ipv6] have been defined to propagate Segment Identifiers (SIDs) for the IPv6 data plane.

SRv6 segment endpoint behaviors have been defined in [RFC8986] and describe how packets should be processed.

This document defines eleven new IPFIX Information Elements (IEs) and three new subregistries within the "IPFIX Information Elements" registry [RFC7012], respectively for the new SRH dimensions, SRv6 endpoint behaviors and routing protocol and PCEP extensions.

2. IPFIX Information Elements

This section defines and describes the new IPFIX IEs.

srhFlagsIPv6
8-bit flags defined in the SRH.

srhTagIPv6
16-bit tag field defined in the SRH that marks a packet as part of a class or group of packets sharing the same set of properties.

srhSegmentIPv6
128-bit IPv6 address that represents an SRv6 segment.

srhActiveSegmentIPv6
128-bit IPv6 address that represents the active SRv6 segment.

srhSegmentIPv6BasicList
Ordered basicList [RFC6313] of zero or more 128-bit IPv6 addresses in the SRH that represents the SRv6 segment list. The Segment List is encoded starting from the last segment of the SR Policy. That is, the first element of the Segment List (Segment List[0]) contains the last segment of the SR Policy, the second element contains the penultimate segment of the SR Policy, and so on.

srhSegmentIPv6ListSection
Exposes the SRH Segment List as defined in section 2 of [RFC8754] as series of n octets.

srhSegmentIPv6sLeft
8-bit unsigned integer defining the number of route segments remaining to reach the end of the segment list.
srhSectionIPv6
Exposes the SRH and its TLV’s as defined in section 2 of [RFC8754] as series of n octets.

srhActiveSegmentIPv6Type
Name of the routing protocol or PCEP extension from where the active SRv6 segment has been learned from.

srhSegmentLocatorLength
The number of significant bits. Together with srhSegmentIPv6 it enables the calculation of the SRv6 Locator.

srhSegmentEndpointBehavior
16-bit unsigned integer that represents a SRv6 Endpoint behavior.

Note that the srhSegmentIPv6, srhSegmentLocatorLength, and srhSegmentEndpointBehavior IPFIX IEs are generic fields, to be used in the context of IPFIX Options Templates or IPFIX Structured Data [RFC6313].

3. Use Cases

By using srhSegmentIPv6BasicList(TBD5) or the srhSegmentIPv6ListSection (TBD6), srhActiveSegmentIPv6 (TBD4), srhSegmentIPv6sLeft (TBD7), srhActiveSegmentIPv6Type(TBD9), the forwardingStatus(89), and some counters information, it is possible to answer the following questions (amongst others):

* how many packets are forwarded or dropped
* if dropped, for which reasons,
* identify the active segment and its control plane protocol,
* the SRv6 segment list,
* the next SRv6 node and its type,
* and how many SRv6 segments are left.

4. IANA Considerations

This document requests IANA to create new IEs (see table 1) and three new subregistries called "IPFIX IPv6 SRH Flags" (table 2), "IPFIX IPv6 SRH Segment type" (table 3) and "IPFIX SRV6 Endpoint Behavior" (table 4) under the "IPFIX Information Elements" registry [RFC7012] available at [IANA-IPFIX] and assign the following initial code points.
<table>
<thead>
<tr>
<th>Element ID</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>TBD1</td>
<td>srhFlagsIPv6</td>
</tr>
<tr>
<td>TBD2</td>
<td>srhTagIPv6</td>
</tr>
<tr>
<td>TBD3</td>
<td>srhSegmentIPv6</td>
</tr>
<tr>
<td>TBD4</td>
<td>srhActiveSegmentIPv6</td>
</tr>
<tr>
<td>TBD5</td>
<td>srhSegmentIPv6BasicList</td>
</tr>
<tr>
<td>TBD6</td>
<td>srhSegmentIPv6ListSection</td>
</tr>
<tr>
<td>TBD7</td>
<td>srhSegmentIPv6sLeft</td>
</tr>
<tr>
<td>TBD8</td>
<td>srhSectionIPv6</td>
</tr>
<tr>
<td>TBD9</td>
<td>srhActiveSegmentIPv6Type</td>
</tr>
<tr>
<td>TBD10</td>
<td>srhSegmentLocatorLength</td>
</tr>
<tr>
<td>TBD11</td>
<td>srhSegmentEndpointBehavior</td>
</tr>
</tbody>
</table>

Table 1: Creates IEs in the "IPFIX Information Elements" registry

Note to the RFC-Editor:

* Please replace TBD1 - TBD16 with the values allocated by IANA
* Please replace the [RFC-to-be] with the RFC number assigned to this document
4.1. srhFlagsIPv6

Name: srhFlagsIPv6 ElementID: TBD1 Description: This Information Element identifies the 8-bit flags defined in the SRH. Values for this Information Element are listed in the "IPFIX IPv6 SRH Flags" registry, see Abstract Data Type: unsigned8 Data Type Semantics: flags Reference: [RFC-to-be], RFC8754[IANA-IPFIX]. srhFlagsIPv6 values must not be directly added to this "IPFIX IPv6 SRH Flags" registry. They must instead be added to the "Segment Routing Header Flags" registry. Both the "IPFIX IPv6 SRH Flags" and the "Segment Routing Header Flags" registries must be kept in synch. Initial values in the registry are defined by the table below.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-1</td>
<td>Unassigned</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>O-flag</td>
<td>[RFC-ietf-6man-spring-srv6-oam-13]</td>
</tr>
<tr>
<td>3-7</td>
<td>Unassigned</td>
<td></td>
</tr>
</tbody>
</table>

Table 2: "IPFIX IPv6 SRH Flags" registry

4.2. srhTagIPv6

Name: srhTagIPv6 ElementID: TBD2 Description: This Information Element identifies the 16-bit tag field defined in the SRH that marks a packet as part of a class or group of packets sharing the same set of properties. Abstract Data Type: unsigned16 Data Type Semantics: identifier Reference: [RFC-to-be], RFC8754

4.3. srhSegmentIPv6

Name: srhSegmentIPv6 ElementID: TBD3 Description: This Information Element identifies the 128-bit IPv6 address that represents an SRv6 segment. Abstract Data Type: ipv6address Data Type Semantics: default Reference: [RFC-to-be], RFC8754

4.4. srhActiveSegmentIPv6

Name: srhActiveSegmentIPv6 ElementID: TBD4 Description: This Information Element identifies the 128-bit IPv6 address that represents the active SRv6 segment. Abstract Data Type: ipv6address Data Type Semantics: default Reference: [RFC-to-be], RFC8754
4.5. srhSegmentIPv6BasicList

Name: srhSegmentIPv6BasicList ElementID: TBD5 Description: This Information Element identifies the Ordered basicList [RFC6313] of zero or more 128-bit IPv6 addresses in the SRH that represents the SRv6 segment list. The Segment List is encoded starting from the last segment of the SR Policy. That is, the first element of the Segment List (Segment List[0]) contains the last segment of the SR Policy, the second element contains the penultimate segment of the SR Policy, and so on. Abstract Data Type: basicList Data Type Semantics: list Reference: [RFC-to-be], RFC8754

4.6. srhSegmentIPv6ListSection

Name: srhSegmentIPv6ListSection ElementID: TBD6 Description: Exposes the SRH Segment List as defined in section 2 of Abstract Data Type: octetArray Data Type Semantics: default Reference: [RFC-to-be], RFC8754 as series of n octets.

4.7. srhSegmentIPv6sLeft

Name: srhSegmentIPv6sLeft ElementID: TBD7 Description: This Information Element identifies the 8-bit unsigned integer defining the number of route segments remaining to reach the end of the segment list. Abstract Data Type: unsigned8 Data Type Semantics: quantity Reference: [RFC-to-be], RFC8754

4.8. srhSectionIPv6

Name: srhSectionIPv6 ElementID: TBD8 Description: This Information Element exposes the SRH and its TLV’s as defined in section 2 of Abstract Data Type: octetArray Data Type Semantics: default Reference: [RFC-to-be], RFC8754 as series of n octets.

4.9. srhActiveSegmentIPv6Type

Name: srhActiveSegmentIPv6Type ElementID: TBD9 Description: This Information Element identifies the name of the routing protocol or PCEP extension from where the active SRv6 segment has been learned from. Values for this Information Element are listed in the "IPFIX IPv6 SRH Segment type" registry, see Abstract Data Type: unsigned8 Data Type Semantics: identifier Reference: [RFC-to-be][IANA-IPFIX]. Initial values in the registry are defined by the table below. New assignments of values will be administered by IANA and are subject to Expert Review [RFC8126]. Experts need to check definitions of new values for completeness, accuracy, and redundancy.
<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>TBD12</td>
<td>Unknown</td>
<td>[RFC-to-be]</td>
</tr>
<tr>
<td>TBD13</td>
<td>Path Computation Element</td>
<td>[RFC-to-be], draft-ietf-pce-segment-routing-ipv6</td>
</tr>
<tr>
<td>TBD14</td>
<td>OSPFv3 Segment Routing</td>
<td>[RFC-to-be], draft-li-ospf-ospfv3-srv6-extensions</td>
</tr>
<tr>
<td>TBD15</td>
<td>IS-IS Segment Routing</td>
<td>[RFC-to-be], draft-ietf-lsr-isis-srv6-extensions</td>
</tr>
<tr>
<td>TBD16</td>
<td>BGP Segment Routing Prefix-SID</td>
<td>[RFC-to-be], draft-ietf-bess-srv6-services</td>
</tr>
</tbody>
</table>

Table 3: "IPFIX IPv6 SRH Segment type" subregistry

4.10. srhSegmentLocatorLength

Name: srhSegmentLocatorLength ElementID: TBD10 Description: This Information Element identifies the number of significant bits and together with srhSegmentIPv6 enables the calculation of the SRv6 Locator. Abstract Data Type: unsigned8 Data Type Semantics: default Reference: [RFC-to-be], RFC8986 Section 3.1

4.11. srhSegmentEndpointBehavior

Name: srhSegmentEndpointBehavior ElementID: TBD11 Description: This Information Element identifies the 16-bit SRv6 Endpoint behavior. Values for this Information Element are listed in the "IPFIX SRV6 Endpoint Behavior" registry, see Abstract Data Type: unsigned16 Data Type Semantics: identifier Reference: [RFC-to-be], RFC8986 Section 4[IANA-IPFIX]. srhSegmentEndpointBehavior values must not be directly added to this "IPFIX SRV6 Endpoint Behavior" registry. They must instead be added to the "Segment Routing SRv6 Endpoint Behaviors" registry. Both the "IPFIX SRV6 Endpoint Behavior" and the "Segment Routing SRv6 Endpoint Behaviors" registries must be kept in synch.
5. Operational Considerations

5.1. SRv6 Segment List

The zero or more 128-bit IPv6 addresses in the SRH [RFC8754] can be exported in two different ways, with two different IPFIX IEs:

* srhSegmentIPv6BasicList
* srhSegmentIPv6ListSection

The srhSegmentIPv6BasicList encodes the SID list of IPv6 addresses with a basicList, specified in the IPFIX Structured Data [RFC6313]. This encoding offers the advantage to the data collection that the different IPv6 addresses are already structured as a list, without the need of post processing. However, this method requires some extra processing on the exporter, to realize the BasicList data mapping.

The srhSegmentIPv6ListSection, on the other hand, encodes the list of IPv6 addresses as an octetArray. This doesn’t impose any data flow manipulation on the exporter, facilitating the immediate export. However, the data collection must be able to decode the IPv6 addresses according the SR specifications. Compared to the srhSegmentIPv6BasicList, the srhSegmentIPv6ListSection flow records length is slightly reduced.

It is not expected that an exporter would support both srhSegmentIPv6BasicList and srhSegmentIPv6ListSection at the same time.
5.2. Compressed SRv6 Segment List Decomposition

The SRv6 segment list in the IPFIX IEs srhSegmentIPv6BasicList and srhSegmentIPv6ListSection could contain compressed-SID containers as described in [I-D.ietf-spring-srv6-srh-compression]. The SID endpoint behaviors described in section 4 of [I-D.ietf-spring-srv6-srh-compression] determine wherever the segment list is compressed or not. The SID Locator as described in section 3.1 [RFC8986], determines the common most significant bits.

6. Security Considerations

There exists no significant extra security considerations regarding the allocation of these new IPFIX IEs compared to [RFC7012].

7. Acknowledgements

The authors would like to thank Yao Liu, Paolo Lucente, Eduard Vasilenko, Alex Huang Feng and Bruno Decraene for their review and valuable comments.

8. References

8.1. Normative References


8.2. Informative References
[I-D.ietf-bess-srv6-services]

[I-D.ietf-lsr-isis-srv6-extensions]

[I-D.ietf-pce-segment-routing-ipv6]

[I-D.ietf-spring-srv6-srh-compression]

[I-D.li-lsr-ospfv3-srv6-extensions]

[IANA-IPFIX]

Appendix A. IPFIX Encoding Examples

This appendix represents three different encodings for the newly introduced IEs, for the example values in the table 5. The three different encodings uses the following IEs, respectively: srhSegmentIPv6BasicList, srhSegmentIPv6ListSection, and srhSectionIPv6.

<table>
<thead>
<tr>
<th>SRH Nr</th>
<th>SRH Flags</th>
<th>SRH Tag</th>
<th>Active Segment Type</th>
<th>Segment List</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>123</td>
<td>IS-IS [TBD15]</td>
<td>2001:db8::1, 2001:db8::2, 2001:db8::3</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>456</td>
<td>IS-IS [TBD15]</td>
<td>2001:db8::4, 2001:db8::5</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>789</td>
<td>IS-IS [TBD15]</td>
<td>2001:db8::6</td>
</tr>
</tbody>
</table>

Table 5: three observed SRH headers and their routing protocol

A.1. Template Record and Data Set with Segment Basic List

With this encoding, the examples in Table 5 are represented with the following IEs:

* SR Flags => srhFlagsIPv6
* SR Tag => srhTagIPv6
* Active Segment Type => srhActiveSegmentIPv6Type
* Segment List => srhSegmentIPv6BasicList
Table 6: Template Record with Basic List Encoding Format

In this example, the Template ID is 256, which will be used in the Data Record. The field length for srhSegmentIPv6BasicList is 0xFFFF, which means the length of this IE is variable, and the actual length of this IE is indicated by the List Length field in the basicList format as per [RFC6313].

The data set is represented as follows:

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SET ID = 256</td>
<td>Length = 136</td>
<td></td>
<td></td>
</tr>
<tr>
<td>srhFlagsIPv6 = 0</td>
<td>srhTagIPv6 = 123</td>
<td></td>
<td></td>
</tr>
<tr>
<td>srhActiveSegmentIPv6 = TBD9</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>List Length = 53</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>semantic=ordered</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>srhSegmentIPv6 = TBD3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Field Length = 16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Segment List[0] = 2001:db8::1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Segment List[1] = 2001:db8::2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>srhFlagsIPv6 = 0</td>
<td>srhTagIPv6 = 456</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>srhActiveSegmentIPv6 = TBD15</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>srhSegmentIPv6 = TBD3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Segment List[0] = 2001:db8::4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>srhFlagsIPv6 = 0</td>
<td>srhTagIPv6 = 789</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>srhActiveSegmentIPv6 = TBD15</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>srhSegmentIPv6 = TBD3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
A.2. Template Record and Data Set with Segment List Section

With this encoding, the examples in Table 5 are represented with the following IEs:

* SR Flags => srhFlagsIPv6
* SR Tag => srhTagIPv6
* Active Segment Type => srhActiveSegmentIPv6Type
* Segment List => srhSegmentIPv6List

<table>
<thead>
<tr>
<th>Segment List[0] = 2001:db8::6 ...</th>
</tr>
</thead>
</table>
| +-----------------------------------+
| | .................................. |
| | .................................. |
| +-----------------------------------+
| +-----------------------------------+
| | .................................. |
| | .................................. |
| +-----------------------------------+
| +-----------------------------------+
| | .................................. |
| | .................................. |
| +-----------------------------------+

Table 7: Data Set Encoding Format for Basic List

Table 8: Template Record with Segment List Section Encoding Format

In this example, the Template ID is 257, which will be used in the Data Record. The field length for srhSegmentIPv6ListSection is 0xFFFF, which means the length of this IE is variable.
The data set is represented as follows:

```
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      SET ID = 257       |            Length = 116          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| srhFlagsIPv6 |        srhTagIPv6 = 123       | srhActiveSegme|
| = 0           |                               | ntIPv...=TBD15 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| 0xFFFF     | 2001:db8::1                                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ... | 2001:db8::2 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ... | 2001:db8::3 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      ...      | 2001:db8::5 | srhFlagsIPv6 |        srhTagIPv6 = 456       |
|      ...      |               | = 0           |                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| srhActiveSegme |    0xFFFF     | 2001:db8::4 |
| ntIPv...=TBD15|               |               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...        |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+```
A.3. Template Record and Data Set with SRH Section

With this encoding, the examples in Table 5 are represented with the following IEs:

* SR Flags + SR Tag + Segment List => srhSectionIPv6
* Active Segment Type => srhActiveSegmentIPv6Type

Table 9: Data Set Encoding Format for Segment List Section

| ... | ... |
| +------------------------------+------------------------------|
| ... | ... |
|                               | srhFlagsIPv6=0 | srhTagIPv6... |
| +------------------------------+-------------------------------+|-----------------|
| = 789                         | srhActiveSegment             | 0xFFFF          |
|                               | IPv6Type=TBD15               |                 |
| +------------------------------+-------------------------------+|-----------------|
|                               | 2001:db8::6 ...              |                 |
| +------------------------------+-------------------------------+|-----------------|
|                               | ...                          |                 |
| +------------------------------+-------------------------------+|-----------------|
|                               | ...                          |                 |
| +------------------------------+-------------------------------+|-----------------|
|                               | ...                          |                 |
| +------------------------------+-------------------------------+|-----------------|
|                               | ...                          |                 |
| +------------------------------+-------------------------------+|-----------------|
|                               | ...                          |                 |

Table 10: Template Record with SRH Section Encoding Format

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>+------------------------------+-----------------+</td>
<td>-----------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SET ID = 2</td>
<td>Length = 16</td>
<td></td>
</tr>
<tr>
<td>+------------------------------+-----------------+</td>
<td>-----------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Template ID = 258</td>
<td>Field Count = 2</td>
<td></td>
</tr>
<tr>
<td>+------------------------------+-----------------+</td>
<td>-----------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>srhActiveSegmentIP... = TBD9</td>
<td>Field Length = 1</td>
<td></td>
</tr>
<tr>
<td>+------------------------------+-----------------+</td>
<td>-----------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>srhSectionIPv6 = TBD8</td>
<td>Field Length = 0xFFFF</td>
<td></td>
</tr>
<tr>
<td>+------------------------------+-----------------+</td>
<td>-----------------</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 10: Template Record with SRH Section Encoding Format
In this example, the Template ID is 258, which will be used in the Data Record. The field length for srhSectionIPv6 is 0xFFFF, which means the length of this IE is variable.

The data set is represented as follows:

```
| SET ID = 258 | Length = (*) |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| srhActiveSegmentIPv6Type = TBD15 | 0xFFFF |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Next Header | Hdr Ext Len | Routing Type | Segments Left |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Last Entry | Flags | Tag |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Segment List[0] 2001:db8::1 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ... |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ... |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ... |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Segment List[1] 2001:db8::2 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ... |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ... |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ... |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ... |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ... |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ... |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Optional Type Length Value objects (variable) |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| srhActiveSegmentIPv6Type = TBD15 | 0xFFFF |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Next Header | Hdr Ext Len | Routing Type | Segments Left |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Last Entry | Flags | Tag |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
Table 11: Data Set Encoding Format for SRH Section

(*) The Length must be calculated to include the optional Type Length Value objects.
A.4. Options Template Record and Data Set for SRv6 end point behavior and Locator Length

This appendix provides an SRv6 EndPoint Behavior Options Template example, for the values presented in Table 12. In the Options Template case, the srhEndPointIPv6 Information Element is a Scope field.

<table>
<thead>
<tr>
<th>Entry Nr</th>
<th>SRH End Point IPv6</th>
<th>SRH End Point Behavior</th>
<th>SRH Segment Locator Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2001:db8::1</td>
<td>End [1]</td>
<td>48</td>
</tr>
<tr>
<td>2</td>
<td>2001:db8::4</td>
<td>End with NEXT-CSID [43]</td>
<td>48</td>
</tr>
<tr>
<td>3</td>
<td>2001:db8::6</td>
<td>End.DX6 [16]</td>
<td>48</td>
</tr>
</tbody>
</table>

Table 12: three observed SRv6 End Point Behaviors

Table 13: Template Record with SRH Section Encoding Format

In this example, the Template ID is 259, which will be used in the Data Record.

The data set is represented as follows:
Table 14: Data Set Encoding Format for SRH Section

(*) The Length must be calculated to include the optional Type Length Value objects.

Authors’ Addresses

Thomas Graf
Swisscom
Binzring 17
CH-8045 Zurich
Switzerland
Email: thomas.graf@swisscom.com

Benoit Claise
Huawei
Email: benoit.claise@huawei.com

Pierre Francois
INSA-Lyon
Lyon
France
Email: pierre.francois@insa-lyon.fr