

Workgroup: ALTO WG

Internet-Draft: draft-ietf-alto-oam-yang-05

Published: 23 February 2023

Intended Status: Standards Track

Expires: 27 August 2023

Authors: J. Zhang D. Dhody
 Tongji University Huawei Technologies
 K. Gao R. Schott Q. Ma
 Sichuan University Deutsche Telekom Huawei

A Yang Data Model for OAM and Management of ALTO Protocol

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 set up the ALTO server, create, update and remove ALTO information resources, manage the access control, configure 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 <https://github.com/ietf-wg-alto/draft-ietf-alto-oam-yang>.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <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 August 2023.

Copyright Notice

Copyright (c) 2023 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](#)
- [2. Requirements Language](#)
- [3. Terminology](#)
 - [3.1. Tree Diagrams](#)
 - [3.2. Prefixes in Data Node Names](#)
 - [3.3. Placeholders in reference Statement](#)
- [4. Design Scope and Requirements](#)
 - [4.1. Scope of Data Model for ALTO O&M](#)
 - [4.2. Basic Requirements](#)
 - [4.3. Additional Requirements for Extensibility](#)
 - [4.4. Overview of ALTO O&M Data Model for Reference ALTO Architecture](#)
- [5. Design of ALTO O&M Data Model](#)
 - [5.1. Overview of ALTO O&M Data Model](#)
 - [5.2. Data Model for ALTO Client Operation and Management](#)
 - [5.3. Data Model for Server-level Operation and Management](#)
 - [5.3.1. Data Model for ALTO Server Setup](#)
 - [5.3.2. Data Model for Logging Management](#)
 - [5.3.3. Data Model for ALTO-related Management](#)
 - [5.3.4. Data Model for Security Management](#)
 - [5.4. Data Model for ALTO Server Configuration Management](#)
 - [5.4.1. Data Source Configuration Management](#)
 - [5.4.2. ALTO Information Resources Configuration Management](#)
 - [5.4.3. ALTO Information Resource Access Control Management](#)
- [6. Design of ALTO O&M Statistics Data Model](#)
 - [6.1. Model for ALTO Server Failure Monitoring](#)
 - [6.2. Model for ALTO-specific Performance Monitoring](#)
- [7. ALTO OAM YANG Module](#)
 - [7.1. The ietf-alto Module](#)
 - [7.2. The ietf-alto-stats Module](#)
- [8. Security Considerations](#)
 - [8.1. The "ietf-alto" YANG Module](#)

8.2.	The "ietf-alto-stats" YANG Module
9.	IANA Considerations
10.	References
10.1.	Normative References
10.2.	Informative References
Appendix A.	Example: Extending the ALTO O&M Data Model
A.1.	Example Module for Extended Server Discovery Manners
A.2.	Example Module for Extended Client Authentication Approaches
A.3.	Example Module for Extended Data Sources
A.4.	Example Module for Information Resource Creation Algorithm
	Acknowledgements
	Authors' Addresses

1. Introduction

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 set up the ALTO server, create, update and remove ALTO information resources, manage the access control, configure server discovery, and collect statistical data.

This document only focuses on the common and implementation-agnostic data model for purposes including deploying an ALTO server/client, operating and managing a running ALTO server/client, functionality/capability configuration of ALTO services, and monitoring ALTO-related performance metrics. Any implementation-specific information is not in the scope of this document. [Section 4.1](#) illustrates more details about what is and is not in the scope. [Section 4.2](#) and [Section 4.3](#) define more concrete requirements for the data model.

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 [\[RFC8189\]](#), [\[RFC8895\]](#), [\[RFC8896\]](#), [\[RFC9240\]](#), [\[RFC9241\]](#), and {RFC9275}), the design will also be extensible for future standard extensions (e.g., [\[I-D.ietf-alto-performance-metrics\]](#)).

The detailed design of the data model is illustrated by [Section 5](#) and [Section 6](#). And some examples of how to extend this data model for the specific ALTO server implementations are shown in [Appendix A](#).

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 Maintenance

*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](#).

Prefix	YANG module	Reference
yang	ietf-yang-types	[RFC6991]
inet	ietf-inet-types	[RFC6991]
tcp	ietf-tcp-server	[I-D.ietf-netconf-tcp-client-server]
tls	ietf-tls-server	[I-D.ietf-netconf-tls-client-server]
http	ietf-http-server	[I-D.ietf-netconf-http-client-server]

Table 1: Prefixes and corresponding YANG modules

3.3. Placeholders in reference Statement

Note to the RFC Editor: This section is to be removed prior to publication.

This draft contains placeholder values that need to be replaced with finalized values at the time of publication. This note summarizes all of the substitutions that are needed. No other RFC Editor instructions are specified elsewhere in this document.

Artwork in this document contains shorthand references to drafts in progress. Please apply the following replacements:

*DDDD --> the assigned RFC value for draft-ietf-netconf-tcp-client-server

*FFFF --> the assigned RFC value for draft-ietf-netconf-tls-client-server

*GGGG --> the assigned RFC value for draft-ietf-netconf-http-client-server

*HHHH --> the assigned RFC value for draft-ietf-netconf-netconf-client-server

*IIII --> the assigned RFC value for draft-ietf-netconf-restconf-client-server

*XXXX --> the assigned RFC value for this draft

*YYYY --> the assigned RFC value for draft-ietf-alto-performance-metrics

4. Design Scope and Requirements

4.1. Scope of Data Model for ALTO O&M

Generally, the following items are in the scope of this document:

*Data model for deploying an ALTO server/client.

*Data model for operating and managing a running ALTO server/client.

*Data model for functionality/capability configuration of ALTO services.

*Data model for monitoring ALTO-related performance metrics.

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](#).

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

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 the ALTO server implementation and YANG modules that these server components need to 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 models 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 [Appendix A](#)).

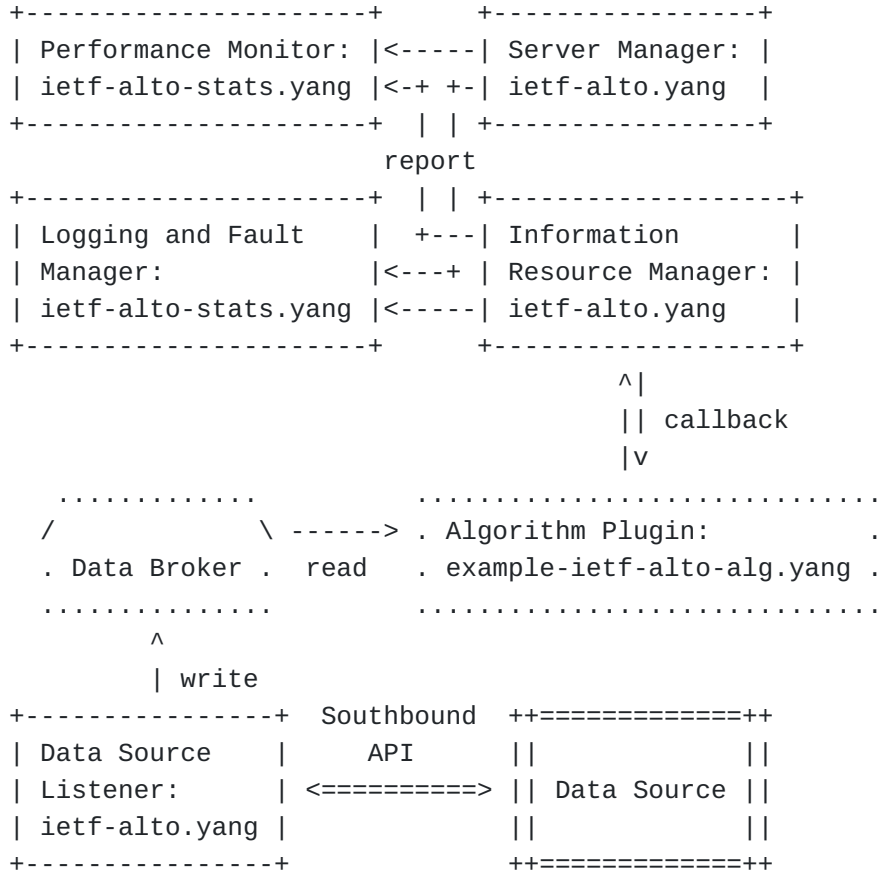


Figure 1: A Reference ALTO Server Architecture and YANG Modules

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 provides all the basic ALTO O&M data models fitting the requirements listed in [Section 4](#).

The top-level container "alto" in the `ietf-alto` module contains a single "alto-server" and multiple "alto-client"s.

The list "alto-client" defines a list of configurations for other applications to launch an ALTO client. They can also be used by data sources and information resource creation algorithms that are configured by an ALTO server instance.

The container "alto-server" contains all the configured and operational parameters of the administrated ALTO server instance.

```

module: ietf-alto
+--rw alto!
  +--rw alto-client* [client-id]
  |   +--rw client-id          string
  |   +--rw server-discovery-client
  |       +---u alto-server-discovery-client-grouping
  +--rw alto-server
  |   +--rw listen
  |       +---u alto-server-listen-stack-grouping
  +--rw server-discovery
  |   +---u alto-server-discovery-grouping
  +--rw logging-system
  |   +---u alto-logging-system-grouping
  +--rw cost-type* [cost-type-name]
  |   +--rw cost-type-name    string
  |   +--rw cost-mode        identityref
  |   +--rw cost-metric      identityref
  |   +--rw description?     string
  |   +--rw cost-context {performance-metrics}?
  |       +--rw cost-source  identityref
  |       +--rw parameters
  |           +--rw (parameters)?
  +--rw meta* [meta-key]
  |   +--rw meta-key          string
  |   +--rw meta-value        string
  +--rw auth-client* [client-id]
  |   +--rw client-id          string
  |   +--rw (authentication)?
  |       +---:(http)
  |           |   +--rw http-auth-client
  |           |       {http-listen,http:client-auth-supported,
  |           |           http:local-users-supported}?
  |           |   +--rw user-id    leafref
  |       +---:(https)
  |           +--rw https-auth-client
  |               {http:client-auth-supported,
  |               http:local-users-supported}?
  |           +--rw user-id    leafref
  +--rw role* [role-name]
  |   +--rw role-name          role-name
  |   +--rw client*
  |       -> /alto/alto-server/auth-client/client-id
  +--rw data-source* [source-id]
  |   +--rw source-id          string
  |   +--rw source-type        identityref
  |   +--rw (update-policy)
  |       |   +---:(reactive)
  |       |       |   +--rw (publish-mode)?
  |       |       |       +---:(on-change)

```



```

| | | | +--rw on-change          empty
| | | | +--:(periodic)
| | | | +--rw feed-interval      uint32
| | | | +--:(proactive)
| | | | +--rw poll-interval      uint32
| | +--rw (source-params)?
+--rw resource* [resource-id]
    +--rw resource-id            resource-id
    +--rw resource-type          identityref
    +--rw description?           string
    +--rw accepted-role*
    | -> /alto/alto-server/role/role-name
    +--rw dependency*
    | -> /alto/alto-server/resource/resource-id
    +--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 filter-costmap-cap
        | +---u algorithm
        +--:(endpointcost)
        | +--rw alto-endpointcost-params
        | +---u endpoint-cost-cap
        | +---u algorithm
        +--:(endpointprop)
        | +--rw alto-endpointprop-params
        | +--rw prop-types*     string
        | +---u algorithm
        +--:(propmap) {propmap}?
        | +--rw alto-propmap-params
        | +---u algorithm
        +--:(cdni) {cdni}?
        | +--rw alto-cdni-params
        | +---u algorithm
        +--:(update) {incr-update}?
        +--rw alto-update-params
        +---u algorithm

```

5.2. Data Model for ALTO Client Operation and Management

The alto-client list contains a list of client-side configurations. server-discovery-client is defined to configure how an ALTO client discovers the ALTO server.

```
module: ietf-alto
  +--rw alto!
    +--rw alto-client* [client-id]
      |  +--rw client-id          string
      |  +--rw server-discovery-client
      |  +---u alto-server-discovery-client-grouping
      ...
```

5.3. Data Model for Server-level Operation and Management

The ALTO server instance contains the following configuration parameters for server-level operation and management for ALTO, which satisfies R1 - R4 in [Section 4.2](#).

```
module: ietf-alto
  +--rw alto!
    ...
    +--rw alto-server
      +--rw listen
        |  +---u alto-server-listen-stack-grouping
      +--rw server-discovery
        |  +---u alto-server-discovery-grouping
      +--rw logging-system
        |  +---u alto-logging-system-grouping
      +--rw cost-type* [cost-type-name]
        |  +--rw cost-type-name    string
        |  +--rw cost-mode         identityref
        |  +--rw cost-metric       identityref
        |  +--rw description?      string
        |  +--rw cost-context {performance-metrics}?
        |    +--rw cost-source     identityref
        |    +--rw parameters
        |      +--rw (parameters)?
      +--rw meta* [meta-key]
        |  +--rw meta-key          string
        |  +--rw meta-value        string
        ...
```

5.3.1. Data Model for ALTO Server Setup

To satisfy R1 in [Section 4.2](#), the ALTO server instance contains the following basic configurations for the server setup.

5.3.1.1. ALTO Server Listen Stack

The "listen" contains all the configurations for the whole server listen stack across HTTP layer, TLS layer and TCP layer.

```
grouping alto-server-grouping:
  +-- base-uri?   inet:uri
grouping alto-server-listen-stack-grouping:
  +-- (transport)
    +--:(http) {http-listen}?
      | +-- http
      |   +-- tcp-server-parameters
      |       | +---u tcp:tcp-server-grouping
      |   +-- http-server-parameters
      |       | +---u http:http-server-grouping
      |   +-- alto-server-parameters
      |       +---u alto-server-grouping
    +--:(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
        |   +---u alto-server-grouping
```

5.3.1.2. ALTO Server Discovery Setup

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 does not contain any configuration for the communication between two ALTO servers. Instead, it provides the configuration for how an ALTO server can be discovered by another ALTO server on demand.

```
grouping alto-server-discovery-grouping:
  +-- (server-discovery-manner)?
    +--:(reverse-dns)
      +-- rdns-naptr-records
        +-- static-prefix*          inet:ip-prefix
        +-- dynamic-prefix-source*
            -> /alto-server/data-source/source-id
```

The server-discovery node provides configuration for ALTO server discovery using different mechanisms. The initial module only defines the reverse-dns case that is used to configure DNS NAPTR

records for ALTO server discovery, which is suggested by [RFC7286] and [RFC8686]. It configures a set of endpoints that can be served by this ALTO server. The node contains two leaf lists. The static list contains a list of manually configured endpoints. The dynamic list points to a list of data sources to retrieve the endpoints dynamically. As suggested by [RFC7286] and [RFC8686], the IP prefixes of the endpoints configured by both static and dynamic lists will be translated into DNS NAPTR resource records for server discovery. The server-discovery-manner choice can be augmented by the future modules to support other mechanisms.

5.3.2. Data Model for Logging Management

To satisfy R2 in [Section 4.2](#), the ALTO server instance contains the following configuration parameters for the logging management.

The logging-system node provides configuration to select a logging system to capture log messages generated by the ALTO server.

By default, syslog is the only supported logging system. When selecting syslog, the related configuration is delegated to the configuration file of the syslog server.

```
grouping alto-logging-system-grouping:
  +-- (logging-system)?
    +--:(syslog)
      +-- syslog-params
        +-- config-file?  inet:uri
```

A specific server implementation can extend the logging-system node to add other logging systems.

5.3.3. Data Model for ALTO-related Management

To satisfy R3 in [Section 4.2](#), the data model contains the following ALTO-related management information.

*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).

5.3.4. Data Model for Security Management

To satisfy R4 in [Section 4.2](#), the data model leverages HTTP and TLS to provide basic security management for an ALTO server. All the related configurations are covered by the server listen stack.

5.4. Data Model for ALTO Server Configuration Management

5.4.1. Data Source Configuration Management

To satisfy R5-1 in [Section 4.2](#), 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 reactively waits the data source for pushing updates. For the proactive update, the ALTO server has to proactively fetch the data source periodically.

To use the reactive update, there are two publish modes:

- *If the on-change attribute is present, the data source is expected to push the update as soon as the data source changes.
- *Otherwise, if the feed-interval attribute is present, the data source is expected to push the updates periodically. The value of feed-interval specifies the interval of pushing the data change updates in milliseconds. If feed-interval is zero, the data source is expected to work in the on-change mode.

To use the proactive update, the poll-interval attribute MUST be present. The value of poll-interval specifies the interval of fetching the data in milliseconds. If poll-interval is zero, the ALTO server will not fetch the data source.

The data-source/source-params node can be augmented for different types of data sources.

```

module: ietf-alto
+--rw alto!
...
+--rw alto-server
...
+--rw data-source* [source-id]
| +--rw source-id                string
| +--rw source-type              identityref
| +--rw (update-policy)
| | +--:(reactive)
| | | +--rw (publish-mode)?
| | | +--:(on-change)
| | | +--rw on-change            empty
| | | +--:(periodic)
| | | +--rw feed-interval        uint32
| | +--:(proactive)
| | +--rw poll-interval          uint32
| +--rw (source-params)?
...

```

This data model only includes common configuration parameters for an ALTO server to correctly interact with a data source. The implementation-specific parameters of any certain data source can be augmented in another module. An example is included in [Appendix A.3](#).

5.4.2. ALTO Information Resources Configuration Management

To satisfy R5-2 and R-3, 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 resources.

Each resource entry provides configurations defining how to create or update an ALTO information resource. Adding a new resource entry notifies the ALTO server to create a new ALTO information resource. Updating an existing resource entry notifies the ALTO server to update the generation parameters (e.g., capabilities and the creation algorithm) of an existing ALTO information resource. Removing an existing resource entry will remove the corresponding ALTO information resource.

A resource entry **MUST** include a unique resource-id and a resource-type.

It can also include an accepted-role node containing a list of role-names that is used by role-based access control for this ALTO information resource. See [Section 5.4.3](#) for details of information resource access control.

For some resource-type, the resource entry MUST also include 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 resource entry MAY also need type-specific parameters. These type-specific parameters include two categories:

1. One category of the type-specific parameters is 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 category of the type-specific parameters is 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 an augmented algorithm node. The augmented algorithm node can reference data sources subscribed by the data-source entries (See [Section 5.4.1](#)). An example of extending the algorithm node for a specific type of resource is included in [Appendix A.4](#).

The developer does not have to 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!
...
+--rw alto-server
...
+--rw resource* [resource-id]
+--rw resource-id resource-id
+--rw resource-type identityref
+--rw description? string
+--rw accepted-role*
| -> /alto/alto-server/role/role-name
+--rw dependency*
| -> /alto/alto-server/resource/resource-id
+--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 filter-costmap-cap
| +---u algorithm
+--:(endpointcost)
| +--rw alto-endpointcost-params
| +---u endpoint-cost-cap
| +---u algorithm
+--:(endpointprop)
| +--rw alto-endpointprop-params
| +--rw prop-types* string
| +---u algorithm
+--:(propmap) {propmap}?
| +--rw alto-propmap-params
| +---u algorithm
+--:(cdni) {cdni}?
| +--rw alto-cdni-params
| +---u algorithm
+--:(update) {incr-update}?
+--rw alto-update-params
+---u algorithm

grouping filter-costmap-cap:
+-- cost-type-names* string
+-- cost-constraints? boolean
+-- max-cost-types? uint32 {multi-cost}?

```



```
+-- testable-cost-type-names*    string {multi-cost}?
+-- calendar-attributes {cost-calendar}?
    +-- cost-type-names*         string
    +-- time-interval-size       decimal64
    +-- number-of-intervals      uint32
grouping endpoint-cost-cap:
    +--u filter-costmap-cap
grouping algorithm:
    +-- (algorithm)
```

5.4.3. ALTO Information Resource Access Control Management

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

The information resource access control is supported using the following configuration:

```
module: ietf-alto
+--rw alto!
  ...
  +--rw alto-server
    ...
    +--rw auth-client* [client-id]
      | +--rw client-id          string
      | +--rw (authentication)?
      |   +--:(http)
      |     | +--rw http-auth-client
      |     |   {http-listen,http:client-auth-supported,
      |     |     http:local-users-supported}?
      |     | +--rw user-id      leafref
      |     +--:(https)
      |       +--rw https-auth-client
      |       |   {http:client-auth-supported,
      |       |     http:local-users-supported}?
      |       | +--rw user-id      leafref
      +--rw role* [role-name]
        | +--rw role-name      role-name
        | +--rw client*
        |   -> /alto/alto-server/auth-client/client-id
        ...
```

It configures the role-based access control:

*auth-client declares a list of ALTO clients that can be authenticated by the internal or external authorization server. This basic model only includes authentication approach directly provided by the HTTP server, but the operators or future documents can augment the authentication choice for different authentication mechanisms.

*role defines a list of roles for access control. Each role contains a list of authenticated ALTO clients. Each client can be assigned to multiple roles. The role-name can be referenced by the accepted-role list of a resource. For a given authenticated

ALTO client, if one of the roles containing it is allowed to access a resource, this client is allowed to access the resource.

6. Design of ALTO O&M Statistics Data Model

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
augment /alto:alto-server/alto:resource/alto:resource-params
  /alto:networkmap/alto:alto-networkmap-params:
  +--ro num-map-pid?           yang:counter32
augment /alto:alto-server/alto:resource/alto:resource-params
  /alto:propmap/alto:alto-propmap-params:
  +--ro num-map-entry?         yang:counter32
augment /alto:alto-server/alto:resource/alto:resource-params
  /alto:cdni/alto:alto-cdni-params:
  +--ro num-base-obj?          yang:counter32
augment /alto:alto-server/alto:resource/alto:resource-params
  /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
```

6.1. Model for ALTO Server Failure Monitoring

To satisfy R6 in [Section 4.2](#), the YANG data module defined in this document contains statistics that indicates server failures.

More specifically, num-total-* and num-total-last-* provides server-level failure counters; num-res-* provides information resource-level failure counters.

6.2. Model for ALTO-specific Performance Monitoring

To satisfy R7 in [Section 4.2](#), 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 of "system and service performance" suggested by [\[RFC7285\]](#) and [\[RFC7971\]](#):

- *Requests and responses for each information resource

- *CPU and memory utilization

- *ALTO map updates

- *Number of PIDs

- *ALTO map sizes

Besides the above measurement information suggested by [\[RFC7285\]](#) and [\[RFC7971\]](#), this data model also contains useful measurement information for other ALTO extensions:

- *num-map-entry and num-base-obj provides measurement for number of generic ALTO entities (for [\[RFC9240\]](#) and [\[RFC9241\]](#))

- *num-upd-sess and num-event-* provides statistics for update sessions and events (for [\[RFC8189\]](#))

This data model only focuses on the performance metrics that can be directly measured at the ALTO server. The following metrics for "measurement of the impact" suggested by [\[RFC7971\]](#) are not contained by this data model:

- *Total amount and distribution of traffic

- *Application performance

7. ALTO OAM YANG Module

7.1. The ietf-alto Module

<CODE BEGINS> file "ietf-alto@2023-02-23.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-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";
  }

  organization
    "IETF ALTO Working Group";

  contact
    "WG Web:  <https://datatracker.ietf.org/wg/alto/about/>
    WG List:  <alto@ietf.org>";

  description
    "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 "2023-02-23" {  
  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.";  
}
```

```
typedef role-name {  
  type string;  
  description  
    "Name of a role for role-based access control.";  
}
```

// 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.";
}

identity cost-source {
  description
    "The cost source attribute indicates the high-level type of the
    data source.";
  reference
    "Section 3.1 of RFC YYYY.";
}

// Identities for ALTO information resources

identity ird {
  base resource-type;
  description
    "Identity for information resource directory.";
}

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

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

identity endpoint-cost {
  base resource-type;
  description

```

```

    "Identity for endpoint cost service.";
}

identity endpoint-prop {
    base resource-type;
    description
        "Identity for endpoint property service.";
}

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

identity cdni {
    base resource-type;
    description
        "Identity for content delivery network interconnection (CDNI)
        advertisement service.";
}

identity update {
    base resource-type;
    description
        "Identity for update stream service.";
}

// 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";
    base cost-metric;
    description
        "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 YYYY";
}

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

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

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

```

```

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

identity tput {
    if-feature "performance-metrics";
    base cost-metric;
    description
        "Section 5.1 of RFC YYYY";
}

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

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

// Identities for cost sources

identity nominal {
    if-feature "performance-metrics";
    base cost-source;
    description
        "The 'nominal' category indicates that the metric value is
        statically configured by the underlying devices.";
    reference
        "Section 3.1 of RFC YYYY";
}

identity sla {
    if-feature "performance-metrics";
    base cost-source;
    description
        "The 'sla' category indicates that the metric value is derived
        from some commitment which this document refers to as
        service-level agreement (SLA).";
    reference
        "Section 3.1 of RFC YYYY";
}

```

```

}

identity estimation {
  if-feature "performance-metrics";
  base cost-source;
  description
    "The 'estimation' category indicates that the metric value is
    computed through an estimation process.";
  reference
    "Section 3.1 of RFC YYYY";
}

// 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.";
  reference
    "Section 8.3.5 of RFC 7285.";
}

feature xdom-disc {
  description
    "Support cross-domain server discovery.";
  reference
    "RFC 8686: Application-Layer Traffic Optimization (ALTO)
    Cross-Domain Server Discovery";
}

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 9275: An Extension for Application-Layer Traffic
        Optimization (ALTO): Path Vector";
}

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

// Groupings

grouping filter-costmap-cap {
    description
        "This grouping defines a 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.";
    }
    description
        "Configuration for CalendarAttributes.";
    reference
        "Section 4.1 of RFC 8896.";
}
}

grouping endpoint-cost-cap {
    uses filter-costmap-cap;
    description
        "This grouping defines EndpointCostCapabilities as the same as
        FilteredCostMapCapabilities defined by the grouping
        filter-costmap-cap.";
    reference
        "Section 11.5.1.4 of RFC 7285";
}

grouping algorithm {
    choice algorithm {
        mandatory true;
        description
            "Information resource creation algorithm to be augmented.";
    }
    description
        "This grouping defines the base data model for information
        resource creation algorithm.";
}

grouping alto-server-grouping {
    description
        "A reuseable grouping for configuring an ALTO server without
        any consideration for how underlying transport sessions are
        established.";
    leaf base-uri {
        type inet:uri;
        description
            "The base URI for the ALTO server.";
    }
}

grouping alto-server-listen-stack-grouping {

```



```

description
  "A reusable grouping for configuring an ALTO server
  'listen' protocol stack for a single connection.";
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;
}
container alto-server-parameters {
    description
        "A wrapper around the ALTO server parameters
        to avoid name collisions.";
    uses alto-server-grouping;
}
}
}
}
}

grouping alto-server-discovery-grouping {
    description
        "Grouping for the configuration of how to set up server
        discovery for clients or other ALTO servers to discovery the
        URI of this ALTO server.";
    choice server-discovery-manner {
        description
            "Selects among available server discovery manners";
        case reverse-dns {
            if-feature xdom-disc;
            description
                "Configure DNS NAPTR records for cross-domain ALTO server
                discovery using reverse DNS lookup.";
            container rdns-naptr-records {
                description

```

```

        "Configuration parameters for DNS NAPTR records.";
    leaf-list static-prefix {
        type inet:ip-prefix;
        description
            "Static IP prefixes in the scope.";
    }
    leaf-list dynamic-prefix-source {
        type leafref {
            path "/alto:alto/alto:alto-server/alto:data-source"
                + "/alto:source-id";
        }
        description
            "Dynamic IP prefixes collected from data sources.";
    }
}
reference
    "RFC 8686: Application-Layer Traffic Optimization (ALTO)
    Cross-Domain Server Discovery.";
}
}

grouping alto-server-discovery-client-grouping {
    description
        "Grouping for configuration of how a client can discover another
        ALTO server.";
    choice server-discovery-client-manner {
        description
            "Selects among available server discovery manners.";
        case reverse-dns {
            if-feature xdom-disc;
            description
                "Use reverse DNS lookup to discover an ALTO server.";
            reference
                "RFC 8686: Application-Layer Traffic Optimization (ALTO)
                Cross-Domain Server Discovery.";
            container rdns-params {
                description
                    "Configuration for reverse DNS lookup.";
                leaf-list dns-server {
                    type inet:host;
                    description
                        "DNS server list for reverse DNS lookup.";
                }
            }
        }
    }
}
}
}

```

```

grouping alto-logging-system-grouping {
  description
    "Grouping for configuration of logging system used by the ALTO
    server.";
  choice logging-system {
    description
      "Selects among available logging systems.";
    case syslog {
      description
        "Use syslog as logging system.";
      container syslog-params {
        description
          "Configuration parameters for syslog.";
        leaf config-file {
          type inet:uri {
            pattern 'file:.*';
          }
          default 'file:/etc/syslog.conf';
          description
            "The file location of the syslog configuration.";
        }
      }
    }
  }
}

// Top-level container

container alto {
  presence "The ALTO is enabled";
  description
    "Parameters for the ALTO client and server.";
  list alto-client {
    key client-id;
    leaf client-id {
      type string;
      description
        "Identifier of a client that can be referenced by a data
        source or a resource creation algorithm to communicate with
        other ALTO servers.";
    }
    container server-discovery-client {
      uses alto-server-discovery-client-grouping;
      description
        "Configuration of how to discover another ALTO server.";
    }
    description
      "The ALTO client configuration.";
  }
}

```

```

container alto-server {
  description
    "The ALTO server instance configuration.";
  container listen {
    description
      "Configure the ALTO server to listen for ALTO clients.";
    uses alto-server-listen-stack-grouping;
  }
  container server-discovery {
    description
      "Configure how the ALTO server to be discovered by others.";
    uses alto-server-discovery-grouping;
  }
  container logging-system {
    description
      "Configure logging system to capture log messages generated
        by the ALTO server.";
    uses alto-logging-system-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";
    }
    leaf description {
      type string;
      description
        "The human-readable description fo the cost-mode and
          cost-mode";
    }
  }
  container cost-context {

```

```

    if-feature "performance-metrics";
    leaf cost-source {
        type identityref {
            base cost-source;
        }
        mandatory true;
        description
            "The referenced cost source";
    }
    container parameters {
        choice parameters {
            description
                "Cases of parameters to be augmented.";
        }
        description
            "Additional computation parameters for the cost
            source.";
    }
    description
        "Context of how the metric is obtained.";
}
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 auth-client {
    key "client-id";
    leaf client-id {
        type string;
        description
            "Identifier to reference an ALTO client.";
    }
}

```

```

choice authentication {
  description
    "Choice of authentication methods to identify this
    ALTO client.";
  case http {
    description
      "The client is authenticated by the HTTP server.";
    container http-auth-client {
      if-feature "http-listen";
      if-feature "http:client-auth-supported";
      if-feature "http:local-users-supported";
      description
        "Parameters of the authenticated HTTP client.";
      leaf user-id {
        type leafref {
          path "/alto:alto/alto:alto-server/alto:listen"
            + "/alto:http/alto:http-server-parameters"
            + "/alto:client-authentication/alto:users"
            + "/alto:user/alto:user-id";
        }
        mandatory true;
        description
          "Reference of the user-id for the authenticated
          client.";
      }
    }
  }
  case https {
    description
      "The client is authenticated by the HTTP server.";
    container https-auth-client {
      if-feature "http:client-auth-supported";
      if-feature "http:local-users-supported";
      description
        "Parameters of the authenticated HTTPS client.";
      leaf user-id {
        type leafref {
          path "/alto:alto/alto:alto-server/alto:listen"
            + "/alto:https/alto:http-server-parameters"
            + "/alto:client-authentication/alto:users"
            + "/alto:user/alto:user-id";
        }
        mandatory true;
        description
          "Reference of the user-id for the authenticated
          client.";
      }
    }
  }
}

```

```

    }
    description
        "List of authenticated ALTO clients.";
}
list role {
    key "role-name";
    leaf role-name {
        type role-name;
        description
            "Name of a role for access control.";
    }
    leaf-list client {
        type leafref {
            path "/alto:alto/alto:alto-server/alto:auth-client"
                + "/alto:client-id";
        }
        description
            "List of authenticated ALTO clients assigned to the role.";
    }
    description
        "List of roles for access control.";
}
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
            "Identify the type of the data source.";
    }
}
choice update-policy {
    mandatory true;
    case reactive {
        description
            "Configuration for the data source listener to
            reactively subscribe data and wait for updates
            published by the data source.";
        choice publish-mode {
            description
                "Configuration for when the data source publish an
                update.";
        }
    }
}

```



```

    case on-change {
      description
        "The data source is requested to publish an update
        once the data has a change.";
      leaf on-change {
        type empty;
        mandatory true;
        description
          "Indicate an on-change subscription.";
      }
    }
  case periodic {
    description
      "The data source is requested to periodically
      publish an update.";
    leaf feed-interval {
      type uint32;
      mandatory true;
      description
        "Duration of time that should occur between
        periodic push updates, in units of seconds.";
    }
  }
}
case proactive {
  description
    "Configuration for the data source listener to
    proactively pull data from the data source.";
  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 {
  description
    "Data source specific configuration.";
}
description
  "List of subscribed data sources.";
}
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-role {
    type leafref {
        path "/alto:alto/alto:alto-server/alto:role"
            + "/alto:role-name";
    }
    description
        "Roles allowed to access this information resource.";
}
leaf-list dependency {
    type leafref {
        path "/alto:alto/alto:alto-server/alto:resource"
            + "/alto:resource-id";
    }
    description
        "A list of dependent information resources.";
}
choice resource-params {
    description
        "Resource-specific configuration.";
    case ird {
        when 'derived-from-or-self(resource-type, "alto:ird)';
        container alto-ird-params {
            leaf delegation {
                type inet:uri;
                mandatory true;
                description
                    "Upstream IRD to be delegated.";
            }
            description
                "IRD-specific configuration.";
        }
    }
}
}

```

```

case networkmap {
  when 'derived-from-or-self(resource-type, '
    + '"alto:network-map")';
  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 {
  when 'derived-from-or-self(resource-type, '
    + '"alto:cost-map")';
  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 filter-costmap-cap;
    uses algorithm;
  }
}

case endpointcost {
  when 'derived-from-or-self(resource-type, '
    + '"alto:endpoint-cost")';
  container alto-endpointcost-params {
    description
      "Endpoint Cost Service specific configuration.";
    reference
      "Section 11.5 of RFC 7285.";
    uses endpoint-cost-cap;
  }
}

```

```

        uses algorithm;
    }
}
case endpointprop {
    when 'derived-from-or-self(resource-type, '
        + '"alto:endpoint-prop")';
    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 {
    when 'derived-from-or-self(resource-type, '
        + '"alto:property-map")';
    if-feature "propmap";
    container alto-propmap-params {
        uses algorithm;
        description
            "(Filtered) Entity Property Map specific
            configuration.";
    }
}
case cdni {
    when 'derived-from-or-self(resource-type, "alto:cdni")';
    if-feature "cdni";
    container alto-cdni-params {
        uses algorithm;
        description
            "CDNi specific configuration";
    }
}
case update {
    when 'derived-from-or-self(resource-type, '
        + '"alto:update")';
    if-feature "incr-update";
    container alto-update-params {
        uses algorithm;
        description
            "Incremental Updates specific configuration";
    }
}

```

```
    }  
  }  
  description  
    "ALTO information resources to be defined";  
  }  
}  
}
```

<CODE ENDS>

7.2. The `ietf-alto-stats` Module

<CODE BEGINS> file "ietf-alto-stats@2023-02-23.yang"

```
module ietf-alto-stats {
  yang-version 1.1;
  namespace
    "urn:ietf:params:xml:ns:yang:ietf-alto-stats";
  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
    "IETF ALTO Working Group";

  contact
    "WG Web:  <https://datatracker.ietf.org/wg/alto/about/>
      WG List: <alto@ietf.org>";

  description
    "This YANG module defines all the statistics 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 "2023-02-23" {
    description
      "Initial Version.";
    reference
```

```

    "RFC XXXX: A YANG Data Model for Operations, Administration,
      and Maintenance of ALTO Protocol.";
}

augment "/alto:alto/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/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/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/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/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/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.";
}
leaf num-event-avg {
    type yang:counter32;
    config false;
    description
        "The average number of update events sent to the connected
        clients.";
}
}
}

```

<CODE ENDS>

8. Security Considerations

8.1. The "ietf-alto" YANG Module

The "ietf-alto" YANG module defines data nodes that are designed to be accessed via YANG based management protocols, such as NETCONF [[RFC6241](#)] and RESTCONF [[RFC8040](#)]. Both of these protocols have mandatory-to-implement secure transport layers (e.g., SSH, TLS) with mutual authentication.

The Network Access Control Model (NACM) [[RFC8341](#)] provides the means to restrict access for particular users to a pre-configured subset of all available protocol operations and content.

None of the readable data nodes in this YANG module are considered sensitive or vulnerable in network environments. The NACM "default-deny-all" extension has not been set for any data nodes defined in this module.

None of the writable data nodes in this YANG module are considered sensitive or vulnerable in network environments. The NACM "default-deny-write" extension has not been set for any data nodes defined in this module.

This module does not define any RPCs, actions, or notifications, and thus the security consideration for such is not provided here.

Please be aware that this module uses groupings defined in other RFCs that define data nodes that do set the NACM "default-deny-all" and "default-deny-write" extensions.

8.2. The "ietf-alto-stats" YANG Module

The "ietf-alto-stats" YANG module defines data nodes that are designed to be accessed via YANG based management protocols, such as NETCONF [[RFC6241](#)] and RESTCONF [[RFC8040](#)]. Both of these protocols have mandatory-to-implement secure transport layers (e.g., SSH, TLS) with mutual authentication.

The Network Access Control Model (NACM) [[RFC8341](#)] provides the means to restrict access for particular users to a pre-configured subset of all available protocol operations and content.

None of the readable data nodes in this YANG module are considered sensitive or vulnerable in network environments. The NACM "default-deny-all" extension has not been set for any data nodes defined in this module.

None of the writable data nodes in this YANG module are considered sensitive or vulnerable in network environments. The NACM "default-

deny-write" extension has not been set for any data nodes defined in this module.

This module does not define any RPCs, actions, or notifications, and thus the security consideration for such is not provided here.

Please be aware that this module uses groupings defined in other RFCs that define data nodes that do set the NACM "default-deny-all" and

9. IANA Considerations

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

URI: urn:ietf:params:xml:ns:yang:ietf-alto
Registrant Contact: The IESG.
XML: N/A; the requested URI is an XML namespace.

URI: urn:ietf:params:xml:ns:yang:ietf-alto-stats
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
Namespace: urn:ietf:params:xml:ns:yang:ietf-alto
Prefix: alto
Reference: [RFCthis]

Name: ietf-alto-stats
Namespace: urn:ietf:params:xml:ns:yang:ietf-alto-stats
Prefix: alto
Reference: [RFCthis]

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

10. References

10.1. Normative References

[[RFC2119](#)] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/

RFC2119, March 1997, <<https://www.rfc-editor.org/rfc/rfc2119>>.

[RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <<https://www.rfc-editor.org/rfc/rfc3688>>.

[RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, DOI 10.17487/RFC6020, October 2010, <<https://www.rfc-editor.org/rfc/rfc6020>>.

[RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 6991, DOI 10.17487/RFC6991, July 2013, <<https://www.rfc-editor.org/rfc/rfc6991>>.

[RFC7285] Alimi, R., Ed., Penno, R., Ed., Yang, Y., Ed., Kiesel, S., Previdi, S., Roome, W., Shalunov, S., and R. Woundy, "Application-Layer Traffic Optimization (ALTO) Protocol", RFC 7285, DOI 10.17487/RFC7285, September 2014, <<https://www.rfc-editor.org/rfc/rfc7285>>.

[RFC7286] Kiesel, S., Stiemerling, M., Schwan, N., Scharf, M., and H. Song, "Application-Layer Traffic Optimization (ALTO) Server Discovery", RFC 7286, DOI 10.17487/RFC7286, November 2014, <<https://www.rfc-editor.org/rfc/rfc7286>>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/rfc/rfc8174>>.

[RFC8177] Lindem, A., Ed., Qu, Y., Yeung, D., Chen, I., and J. Zhang, "YANG Data Model for Key Chains", RFC 8177, DOI 10.17487/RFC8177, June 2017, <<https://www.rfc-editor.org/rfc/rfc8177>>.

[RFC8189] Randriamasy, S., Roome, W., and N. Schwan, "Multi-Cost Application-Layer Traffic Optimization (ALTO)", RFC 8189, DOI 10.17487/RFC8189, October 2017, <<https://www.rfc-editor.org/rfc/rfc8189>>.

[RFC8340] Bjorklund, M. and L. Berger, Ed., "YANG Tree Diagrams", BCP 215, RFC 8340, DOI 10.17487/RFC8340, March 2018, <<https://www.rfc-editor.org/rfc/rfc8340>>.

[RFC8686] Kiesel, S. and M. Stiemerling, "Application-Layer Traffic Optimization (ALTO) Cross-Domain Server Discovery", RFC 8686, DOI 10.17487/RFC8686, February 2020, <<https://www.rfc-editor.org/rfc/rfc8686>>.

[RFC8895]

Roome, W. and Y. Yang, "Application-Layer Traffic Optimization (ALTO) Incremental Updates Using Server-Sent Events (SSE)", RFC 8895, DOI 10.17487/RFC8895, November 2020, <<https://www.rfc-editor.org/rfc/rfc8895>>.

[RFC8896]

Randriamasy, S., Yang, R., Wu, Q., Deng, L., and N. Schwan, "Application-Layer Traffic Optimization (ALTO) Cost Calendar", RFC 8896, DOI 10.17487/RFC8896, November 2020, <<https://www.rfc-editor.org/rfc/rfc8896>>.

10.2. Informative References

[I-D.ietf-alto-performance-metrics]

Wu, Q., Yang, Y. R., Lee, Y., Dhody, D., Randriamasy, S., and L. M. Contreras, "ALTO Performance Cost Metrics", Work in Progress, Internet-Draft, draft-ietf-alto-performance-metrics-28, 21 March 2022, <<https://datatracker.ietf.org/doc/html/draft-ietf-alto-performance-metrics-28>>.

[I-D.ietf-netconf-http-client-server]

Watsen, K., "YANG Groupings for HTTP Clients and HTTP Servers", Work in Progress, Internet-Draft, draft-ietf-netconf-http-client-server-12, 12 December 2022, <<https://datatracker.ietf.org/doc/html/draft-ietf-netconf-http-client-server-12>>.

[I-D.ietf-netconf-netconf-client-server]

Watsen, K., "NETCONF Client and Server Models", Work in Progress, Internet-Draft, draft-ietf-netconf-netconf-client-server-28, 12 December 2022, <<https://datatracker.ietf.org/doc/html/draft-ietf-netconf-netconf-client-server-28>>.

[I-D.ietf-netconf-restconf-client-server]

Watsen, K., "RESTCONF Client and Server Models", Work in Progress, Internet-Draft, draft-ietf-netconf-restconf-client-server-28, 12 December 2022, <<https://datatracker.ietf.org/doc/html/draft-ietf-netconf-restconf-client-server-28>>.

[I-D.ietf-netconf-tcp-client-server]

Watsen, K. and M. Scharf, "YANG Groupings for TCP Clients and TCP Servers", Work in Progress, Internet-Draft, draft-ietf-netconf-tcp-client-server-15, 12 December 2022, <<https://>

datatracker.ietf.org/doc/html/draft-ietf-netconf-tcp-client-server-15>.

[I-D.ietf-netconf-tls-client-server]

Watsen, K., "YANG Groupings for TLS Clients and TLS Servers", Work in Progress, Internet-Draft, draft-ietf-netconf-tls-client-server-32, 12 December 2022, <<https://datatracker.ietf.org/doc/html/draft-ietf-netconf-tls-client-server-32>>.

[RFC2622]

Alaettinoglu, C., Villamizar, C., Gerich, E., Kessens, D., Meyer, D., Bates, T., Karrenberg, D., and M. Terpstra, "Routing Policy Specification Language (RPSL)", RFC 2622, DOI 10.17487/RFC2622, June 1999, <<https://www.rfc-editor.org/rfc/rfc2622>>.

[RFC6241]

Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <<https://www.rfc-editor.org/rfc/rfc6241>>.

[RFC7921]

Atlas, A., Halpern, J., Hares, S., Ward, D., and T. Nadeau, "An Architecture for the Interface to the Routing System", RFC 7921, DOI 10.17487/RFC7921, June 2016, <<https://www.rfc-editor.org/rfc/rfc7921>>.

[RFC7971]

Stiemerling, M., Kiesel, S., Scharf, M., Seidel, H., and S. Previdi, "Application-Layer Traffic Optimization (ALTO) Deployment Considerations", RFC 7971, DOI 10.17487/RFC7971, October 2016, <<https://www.rfc-editor.org/rfc/rfc7971>>.

[RFC8040]

Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <<https://www.rfc-editor.org/rfc/rfc8040>>.

[RFC8341]

Bierman, A. and M. Bjorklund, "Network Configuration Access Control Model", STD 91, RFC 8341, DOI 10.17487/RFC8341, March 2018, <<https://www.rfc-editor.org/rfc/rfc8341>>.

[RFC8342]

Bjorklund, M., Schoenwaelder, J., Shafer, P., Watsen, K., and R. Wilton, "Network Management Datastore Architecture (NMDA)", RFC 8342, DOI 10.17487/RFC8342, March 2018, <<https://www.rfc-editor.org/rfc/rfc8342>>.

[RFC8346]

Clemm, A., Medved, J., Varga, R., Liu, X., Ananthakrishnan, H., and N. Bahadur, "A YANG Data Model

for Layer 3 Topologies", RFC 8346, DOI 10.17487/RFC8346, March 2018, <<https://www.rfc-editor.org/rfc/rfc8346>>.

- [RFC8641] Clemm, A. and E. Voit, "Subscription to YANG Notifications for Datastore Updates", RFC 8641, DOI 10.17487/RFC8641, September 2019, <<https://www.rfc-editor.org/rfc/rfc8641>>.
- [RFC9240] Roome, W., Randriamasy, S., Yang, Y., Zhang, J., and K. Gao, "An Extension for Application-Layer Traffic Optimization (ALTO): Entity Property Maps", RFC 9240, DOI 10.17487/RFC9240, July 2022, <<https://www.rfc-editor.org/rfc/rfc9240>>.
- [RFC9241] Seedorf, J., Yang, Y., Ma, K., Peterson, J., and J. Zhang, "Content Delivery Network Interconnection (CDNI) Footprint and Capabilities Advertisement Using Application-Layer Traffic Optimization (ALTO)", RFC 9241, DOI 10.17487/RFC9241, July 2022, <<https://www.rfc-editor.org/rfc/rfc9241>>.
- [RFC9275] Gao, K., Lee, Y., Randriamasy, S., Yang, Y., and J. Zhang, "An Extension for Application-Layer Traffic Optimization (ALTO): Path Vector", RFC 9275, DOI 10.17487/RFC9275, September 2022, <<https://www.rfc-editor.org/rfc/rfc9275>>.

Appendix A. Example: Extending the ALTO O&M Data Model

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 server-discovery-manner choice of the server-discovery.
- *The authentication choice of each auth-client.
- *The data-source choice.
- *The algorithm choice of the resource-params of each resource.

A.1. Example Module for Extended Server Discovery Manners

The base data model defined by `ietf-alto.yang` only includes a reverse DNS based server discovery manner. The following example module demonstrates how additional server discovery manners can be augmented into the base data model.

The case `internet-routing-registry` allows the ALTO server to update the server URI to the attribute of the corresponding aut-num class in IRR.

The case `peeringdb` allows the ALTO server to update the server URI to the org object of the organization record in PeeringDB.

```

module example-ietf-alto-server-discovery {
  yang-version 1.1;

  namespace "urn:example:ietf-alto-server-discovery";
  prefix "alto-disc";

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

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

  organization
    "IETF ALTO Working Group";

  contact
    "WG Web:  <https://datatracker.ietf.org/wg/alto/about/>
    WG List:  <alto@ietf.org>";

  description
    "This YANG module defines an example of the extended ALTO server
    discovery manners for IRR and PeeringDB.

    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 "2023-02-10" {
    description
      "Initial Version.";
    reference
      "RFC XXXX: A YANG Data Model for Operations, Administration,
      and Maintenance of ALTO Protocol.";
  }
}

```

```

}

augment "/alto:alto/alto:alto-server/alto:server-discovery"
  + "/alto:server-discovery-manner" {
  description
    "Examples of server discovery mechanisms provided by the ALTO
    server.";
  case internet-routing-registry {
    description
      "Update descr attributes of a aut-num class in a Internet
      Routing Registry (IRR) database for ALTO server discovery
      using RPSL.";
    reference
      "RFC 2622: Routing Policy Specification Language (RPSL).";
    container irr-params {
      description
        "Configuration parameters for IRR database.";
      leaf aut-num {
        type inet:as-number;
        description
          "The autonomous system (AS) to be updated.";
      }
    }
  }
}

case peeringdb {
  description
    "Update metadata of a network record in PeeringDB database
    for ALTO server discovery using PeeringDB lookup.";
  container peeringdb-params {
    description
      "Configuration parameters for PeeringDB database.";
    leaf org-id {
      type uint32;
      description
        "The ID referring to the org object of the
        organization record in PeeringDB.";
    }
  }
}

}

augment "/alto:alto/alto:alto-client/alto:server-discovery-client"
  + "/alto:server-discovery-client-manner" {
  description
    "Examples of server discovery mechanisms used by the ALTO
    client.";
  case internet-routing-registry {
    description
      "Use Internet Routing Registry (IRR) to discover an ALTO

```

```

        server.";
reference
    "RFC 2622: Routing Policy Specification Language (RPSL).";
container irr-params {
    description
        "Configuration for IRR query using RPSL.";
    leaf whois-server {
        type inet:host;
        description
            "Whois server for IRR query using RPSL.";
    }
}
}
case peeringdb {
    description
        "Use PeeringDB to discover an ALTO server.";
    container peeringdb-params {
        description
            "Configuration for PeeringDB query";
        leaf peeringdb-endpoint {
            type inet:uri;
            description
                "Endpoint of PeeringDB API server.";
        }
    }
}
}
}
}

```

A.2. Example Module for Extended Client Authentication Approaches

The base data model defined by `ietf-alto.yang` only includes the client authentication approaches directly provided by the HTTP server. However, a real implementation may authenticate clients in different ways, e.g., it may delegate the authentication to a third-party OAuth 2.0 server. The following example module demonstrates how additional client authentication approaches can be augmented into the base data model.

In this example, the case `oauth2` includes the URI to a third-party OAuth 2.0 based authorization server that the ALTO server can redirect to for the client authentication.

```

module example-ietf-alto-auth {
  yang-version 1.1;

  namespace "urn:example:ietf-alto-auth";
  prefix "alto-auth";

  import ietf-inet-types {
    prefix "inet";
    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
    "IETF ALTO Working Group";

  contact
    "WG Web:  <https://datatracker.ietf.org/wg/alto/about/>
    WG List:  <alto@ietf.org>";

  description
    "This YANG module defines an example of the extended ALTO client
    authentication approaches for the role-based access control.

    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 "2023-02-10" {
    description
      "Initial Version.";
    reference
      "RFC XXXX: A YANG Data Model for Operations, Administration,
      and Maintenance of ALTO Protocol.";
  }

```

```

}

augment "/alto:alto/alto:alto-server/alto:auth-client"
+ "/alto:authentication" {
  description
    "Example of extended ALTO client authentication approaches.";
  case oauth2 {
    description
      "Example of authentication by a third-party OAuth 2.0
      server.";
    container oauth2 {
      description
        "Parameters for authentication by a third-party OAuth 2.0
        server.";
      leaf oauth2-server {
        type inet:uri;
        description
          "The URI to the authorization server.";
      }
    }
  }
}
}
}
}

```


A.3. Example Module for Extended Data Sources

The base data model defined by `ietf-alto.yang` does not include any choice cases for specific data sources. The following example module demonstrates how a implementation-specific data source can be augmented into the base data model.

The `yang-datastore` case is used to import the YANG data from a YANG model-driven datastore. It includes:

- *`datastore` to indicate which datastore is fetched.
- *`target-paths` to specify the list of nodes or subtrees in the datastore.
- *`protocol` to indicate which protocol is used to access the datastore. Either `restconf` or `netconf` can be used.

```

module example-ietf-alto-data-source {
  yang-version 1.1;

  namespace "urn:example:ietf-alto-data-source";
  prefix "alto-ds";

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

  import ietf-datastores {
    prefix ds;
    reference
      "RFC8342: Network Management Datastore Architecture (NMDA)";
  }

  import ietf-yang-push {
    prefix yp;
    reference
      "RFC8641: Subscription to YANG Notifications for Datastore
      Updates";
  }

  import ietf-netconf-client {
    prefix ncc;
    reference
      "RFC HHHH: NETCONF Client and Server Models";
  }

  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/>
    WG List:  <alto@ietf.org>";

  description
    "This YANG module defines an example of the extended ALTO data
    source for YANG-based datastore."

```

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 "2023-02-10" {
  description
    "Initial Version.";
  reference
    "RFC XXXX: A YANG Data Model for Operations, Administration,
    and Maintenance of ALTO Protocol.";
}

identity yang-datastore {
  base alto:source-type;
  description
    "Identity for data source of YANG-based datastore.";
}

identity protocol-type {
  description
    "Base identity for protocol type.";
}

identity netconf {
  base protocol-type;
  description
    "Identity for NETCONF protocol.";
}

identity restconf {
  base protocol-type;
  description
    "Identity for RESTCONF protocol.";
}

augment "/alto:alto/alto:alto-server/alto:data-source"
  + "/alto:source-params" {
  description
    "Example of data source for YANG datastore.";
  case yang-datastore {
```

```

when 'derived-from-or-self(source-type, "alto-ds:yang-datastore")'
description
  "Example data source for local and/or remote YANG datastore.";
container yang-datastore-source-params {
  description
    "YANG datastore specific configuration.";
  leaf datastore {
    type ds:datastore-ref;
    mandatory true;
    description
      "Identity reference of the datastore from which to get
        data.";
  }
  list target-paths {
    key name;
    description
      "XPath to subscribed YANG datastore node or subtree.";
    leaf name {
      type string;
      description
        "Identifier of the supported xpath or subtree filters.";
    }
    uses yp:selection-filter-types;
  }
  leaf protocol {
    type identityref {
      base protocol-type;
    }
    description
      "Protocol used to access the YANG datastore.";
  }
  container restconf {
    uses rcc:restconf-client-app-grouping {
      when 'derived-from-or-self(..../protocol, "restconf")';
    }
    description
      "Parameters for restconf endpoint of the YANG datastore.";
  }
  container netconf {
    uses ncc:netconf-client-app-grouping {
      when 'derived-from-or-self(..../protocol, "netconf")';
    }
    description
      "Parameters for netconf endpoint of the YANG datastore.";
  }
}
}
}
}

```

A.4. Example Module for Information Resource Creation Algorithm

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

augment /alto:alto/alto:alto-server/alto:resource
    /alto:resource-params/alto:networkmap
    /alto:alto-networkmap-params/alto:algorithm:
+--:(l3-unicast-cluster)
+--rw l3-unicast-cluster-algorithm
+--rw l3-unicast-topo      leafref
+--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 target path name of an operational yang-datastore data source node (See [Appendix A.3](#)) subscribed in the data-source list (See [Section 5.4.1](#)). The referenced target path in the corresponding yang-datastore data source is assumed 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 5.4.1](#)).

```

module example-ietf-alto-alg {
  yang-version 1.1;

  namespace "urn:example:ietf-alto-alg";
  prefix "alto-alg";

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

  import ietf-datastores {
    prefix ds;
    reference
      "RFC8342: Network Management Datastore Architecture (NMDA)";
  }

  import example-ietf-alto-data-source {
    prefix "alto-ds";
  }

  organization
    "IETF ALTO Working Group";

  contact
    "WG Web:  <https://datatracker.ietf.org/wg/alto/about/>
    WG List:  <alto@ietf.org>";

  description
    "This YANG module defines an example of the extended ALTO
    information resource creation algorithm for translating an L3
    unicast topology of I2RS to an ALTO network map.

    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 "2023-02-10" {

```

```

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

augment "/alto:alto/alto:alto-server/alto:resource"
    + "/alto:resource-params/alto:networkmap"
    + "/alto:alto-networkmap-params/alto:algorithm" {
    description
        "Example of network map creation algorithm.";
    case l3-unicast-cluster {
        description
            "Example algorithm translating an L3 unicast topology of I2RS
            to an ALTO network map";
        container l3-unicast-cluster-algorithm {
            description
                "Parameters for l3-unicast-cluster algorithm";
            leaf l3-unicast-topo {
                type leafref {
                    path "/alto:alto/alto:alto-server/alto:data-source"
                        + "/alto-ds:yang-datastore-source-params"
                        + "/alto-ds:target-paths/alto-ds:name";
                }
                must 'deref(.)/../..'
                    + '/alto-ds:datastore = "ds:operational"';
                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 Qin Wu for the help with drafting the initial version of the YANG modules. Thanks also to Adrian Farrel, Jordi Ros Giralt, Qiao Xiang, and Qin Wu 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

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore 560066
Karnataka
India

Email: dhruv.ietf@gmail.com

Kai Gao
Sichuan University
No.24 South Section 1, Yihuan Road
Chengdu
Sichuan, 610000
China

Email: kaigao@scu.edu.cn

Roland Schott
Deutsche Telekom
Heinrich-Hertz-Strasse 3-7
64295 Darmstadt
Germany

Email: Roland.Schott@telekom.de

Qiufang Ma
Huawei
101 Software Avenue, Yuhua District
Nanjing
Jiangsu, 210012
China

Email: maqiufang1@huawei.com