Stream: Internet Engineering Task Force (IETF)
RFC: 9241
Category: Standards Track
Published: July 2022
ISSN: 2070-1721
Authors: J. Seedorf, *HFT Stuttgart*; Y. Yang, *Yale University*; K. Ma, *Ericsson*; J. Peterson, *NeuStar*; J. Zhang, *Tongji University*

RFC 9241
Content Delivery Network Interconnection (CDNI) Footprint and Capabilities Advertisement Using Application-Layer Traffic Optimization (ALTO)

Abstract

The Content Delivery Networks Interconnection (CDNI) framework in RFC 6707 defines a set of protocols to interconnect CDNs to achieve multiple goals, including extending the reach of a given CDN. A CDNI Request Routing Footprint & Capabilities Advertisement interface (FCI) is needed to achieve the goals of a CDNI. RFC 8008 defines the FCI semantics and provides guidelines on the FCI protocol, but the exact protocol is not specified. This document defines a new Application-Layer Traffic Optimization (ALTO) service, called "CDNI Advertisement Service", that provides an implementation of the FCI, following the guidelines defined in RFC 8008.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc9241.

Copyright Notice

Table of Contents

1. Introduction

The ability to interconnect multiple content delivery networks (CDNs) has many benefits, including increased coverage, capability, and reliability. The Content Delivery Networks Interconnection (CDNI) framework [RFC6707] defines four interfaces to interconnect CDNs: (1) the CDNI Request Routing Interface, (2) the CDNI Metadata Interface, (3) the CDNI Logging Interface, and (4) the CDNI Control Interface.

Among these four interfaces, the CDNI Request Routing Interface provides key functions, as specified in [RFC6707]:

> The CDNI Request Routing interface enables a Request Routing function in an Upstream CDN to query a Request Routing function in a Downstream CDN to determine if the Downstream CDN is able (and willing) to accept the delegated Content Request. It also allows the Downstream CDN to control what should be returned to the User Agent in the redirection message by the upstream Request Routing function.

At a high level, therefore, the scope of the CDNI Request Routing Interface contains two main tasks: (1) determining if the dCDN (downstream CDN) is willing to accept a delegated Content Request and (2) redirecting the Content Request coming from a uCDN (upstream CDN) to the proper entry point or entity in the dCDN.

Correspondingly, the Request Routing Interface is broadly divided into two functionalities: (1) the CDNI Footprint & Capabilities Advertisement interface (FCI) defined in [RFC8008] and (2) the CDNI Request Routing Redirection interface (RI) defined in [RFC7975]. This document focuses on the first functionality (CDNI FCI).

Specifically, CDNI FCI allows both an Advertisement from a dCDN to a uCDN (push) and a query from a uCDN to a dCDN (pull) so that the uCDN knows whether it can redirect a particular user request to that dCDN.

A key component in defining the CDNI FCI is defining the objects that describe the footprints and capabilities of a dCDN. Such objects are already specified in Section 5 of [RFC8008]. However, no protocol is defined to transport and update such objects between a uCDN and a dCDN.

To define such a protocol, this document specifies an extension of the Application-Layer Traffic Optimization (ALTO) Protocol [RFC7285] by introducing a new ALTO service called "CDNI Advertisement Service".

Section 2.3 discusses the benefits in using ALTO as a transport protocol.

2. Terminology and Background

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.

The design of CDNI FCI transport using ALTO assumes an understanding of both FCI semantics and ALTO. Hence, this document starts with a non-normative review of both.

2.1. Terminology

The document uses the CDNI terms defined in [RFC6707], [RFC8006], and [RFC8008]. Also, the document uses the ALTO terms defined in [RFC7285] and [RFC9240]. This document uses the following abbreviations:

ALTO: Application-Layer Traffic Optimization

ASN: Autonomous System Number

CDN: Content Delivery Network

CDNI: CDN Interconnection

dCDN: Downstream CDN

FCI: CDNI FCI, CDNI Request Routing Footprint & Capabilities Advertisement interface

IRD: Information Resource Directory in ALTO

PID: Provider-defined Identifier in ALTO

uCDN: Upstream CDN

2.2. Semantics of FCI Advertisement

[RFC8008] defines the semantics of CDNI FCI, provides guidance on what footprint and capabilities mean in a CDNI context, and specifies the requirements on the CDNI FCI transport protocol. The definitions in [RFC8008] depend on [RFC8006]. Below is a non-normative review of key related points of [RFC8008] and [RFC8006]. For detailed information and normative specification, the reader should refer to these two RFCs.

- Multiple types of mandatory-to-implement footprints (i.e., "ipv4cidr", "ipv6cidr", "asn", and "countrycode") are defined in [RFC8006]. A "set of IP prefixes" can contain both full IP addresses (i.e., a /32 for IPv4 or a /128 for IPv6) and IP prefixes with an arbitrary prefix length. There must also be support for multiple IP address versions, i.e., IPv4 and IPv6, in such a footprint.

- Multiple initial types of capabilities are defined in [RFC8008] including (1) Delivery Protocol, (2) Acquisition Protocol, (3) Redirection Mode, (4) capabilities related to CDNI Logging, and (5) capabilities related to CDNI Metadata. They are required in all cases and, therefore, considered as mandatory-to-implement capabilities for all CDNI FCI implementations.
- Footprint and capabilities are defined together and cannot be interpreted independently from each other. Specifically, [RFC8008] integrates footprint and capabilities with an approach of "capabilities with footprint restrictions", by expressing capabilities on a per footprint basis.
- Specifically, for all mandatory-to-implement footprint types, footprints can be viewed as constraints for delegating requests to a dCDN: a dCDN footprint advertisement tells the uCDN the limitations for delegating a request to the dCDN. For IP prefixes or Autonomous System Numbers (ASNs), the footprint signals to the uCDN that it should consider the dCDN a candidate only if the IP address of the Request Routing source falls within the prefix set or ASN, respectively. The CDNI specifications do not define how a given uCDN determines what address ranges are in a particular ASN. Similarly, for country codes, a uCDN should only consider the dCDN a candidate if it covers the country of the Request Routing source. The CDNI specifications do not define how a given uCDN determines the country of the Request Routing source. Different types of footprint constraints can be combined together to narrow the dCDN candidacy, i.e., the uCDN should consider the dCDN a candidate only if the request routing source satisfies all the types of footprint constraints in the advertisement.
- Given that a large part of Footprint and Capabilities Advertisement may happen in contractual agreements, the semantics of CDNI Footprint and Capabilities Advertisement refers to answering the following question: what exactly still needs to be advertised by the CDNI FCI? For instance, updates about temporal failures of part of a footprint can be useful information to convey via the CDNI FCI. Such information would provide updates on information previously agreed to in contracts between the participating CDNs. In other words, the CDNI FCI is a means for a dCDN to provide changes and updates regarding a footprint and/or capabilities that it has previously agreed to serve in a contract with a uCDN. Hence, server push and incremental encoding will be necessary techniques.

2.3. ALTO Background and Benefits

Application-Layer Traffic Optimization (ALTO) [RFC7285] defines an approach for conveying network-layer (topology) information to "guide" the resource provider selection process in distributed applications that can choose among several candidate resources providers to retrieve a given resource. Usually, it is assumed that an ALTO server conveys information that these applications cannot measure or have difficulty measuring themselves [RFC5693].

Originally, ALTO was motivated by optimizing cross-ISP traffic generated by peer-to-peer applications [RFC5693]. However, ALTO can also be used for improving the Request Routing in CDNs. In particular, Section 5 of [RFC7971] explicitly mentions ALTO as a candidate protocol to improve the selection of a CDN surrogate or origin.

The following reasons make ALTO a suitable candidate protocol for dCDN selection as part of CDNI Request Routing and, in particular, for an FCI protocol:

- Application-Layer-oriented: ALTO is a protocol specifically designed to improve application-layer traffic (and application-layer connections among hosts on the Internet) by providing additional information to applications that these applications could not easily retrieve themselves. This matches the need of CDNI, where a uCDN wants to improve application-layer CDN request routing by using information (provided by a dCDN) that the uCDN could not easily obtain otherwise. Hence, ALTO can help a uCDN to select a proper dCDN by first providing dCDNs' capabilities as well as footprints (see Section 3) and then providing costs of surrogates in a dCDN by ALTO cost maps.
- Security: The identification between uCDNs and dCDNs is an important requirement (see Section 8). ALTO maps can be signed and hence provide inherent origin protection. Please see Section 15.1.2 of [RFC7285] for detailed protection strategies.
- RESTful design: The ALTO Protocol has undergone extensive revisions in order to provide a RESTful design regarding the client-server interaction specified by the protocol. It is flexible and extensible enough to handle existing and potential future data formats defined by CDNI. It can provide the consistent client-server interaction model for other existing CDNI interfaces or potential future extensions and therefore reduce the learning cost for both users and developers, although they are not in the scope of this document. A CDNI FCI interface based on ALTO would inherit this RESTful design. Please see Section 3.
- Error handling: The ALTO Protocol provides extensive error handling in the whole request and response process (see Section 8.5 of [RFC7285]). A CDNI FCI interface based on ALTO would inherit this extensive error-handling framework. Please see Section 5.
- Map Service: The semantics of an ALTO network map is an exact match for the needed information to convey a footprint by a dCDN, in particular, if such a footprint is being expressed by IP prefix ranges. Please see Section 4.
- Filtered Map Service: The ALTO map filtering service would allow a uCDN to query only for parts of an ALTO map. For example, the ALTO filtered property Map Service can enable a uCDN to query properties of a part of footprints efficiently. Please see Section 6.
- Server-initiated notifications and incremental updates: When the footprint or the capabilities of a dCDN change (i.e., unexpectedly from the perspective of a uCDN), server-initiated notifications would enable a dCDN to inform a uCDN about such changes directly. Consider the case where -- due to failure -- part of the footprint of the dCDN is not functioning, i.e., the CDN cannot serve content to such clients with reasonable QoS. Without server-initiated notifications, the uCDN might still use a recent network and cost map from the dCDN and therefore redirect requests to the dCDN that it cannot serve. Similarly, the possibility for incremental updates would enable efficient conveyance of the aforementioned (or similar) status changes by the dCDN to the uCDN. The newest design of ALTO supports server-pushed incremental updates [RFC8895].
- Content availability on hosts: A dCDN might want to express CDN capabilities in terms of certain content types (e.g., codecs and/or formats, or content from certain content providers). ALTO Entity Property Map [RFC9240] would enable a dCDN to make such

information available to a uCDN. This would enable a uCDN to assess whether a dCDN has the capabilities for a given type of content requested.

- Resource availability on hosts or links: The capabilities on links (e.g., maximum bandwidth) or caches (e.g., average load) might be useful information for a uCDN for optimized dCDN selection. For instance, if a uCDN receives a streaming request for content with a certain bitrate, it needs to know if it is likely that a dCDN can fulfill such stringent application-level requirements (i.e., can be expected to have enough consistent bandwidth) before it redirects the request. In general, if ALTO could convey such information via ALTO Entity Property Map [RFC9240], it would enable more sophisticated means for dCDN selection with ALTO. The ALTO Path Vector extension [ALTO-PATH-VECTOR] is designed to allow ALTO clients to query information such as capacity regions for a given set of flows.

3. CDNI Advertisement Service

The ALTO Protocol relies upon the ALTO information service framework, which consists of multiple services. All ALTO services are "provided through a common transport protocol; messaging structure and encoding; and transaction model" [RFC7285]. The ALTO Protocol specification defines multiple initial services, e.g., the ALTO Network Map Service and Cost Map Service.

This document defines a new ALTO service, called "CDNI Advertisement Service", which conveys JSON [RFC8259] objects of media type "application/alto-cdni+json". These JSON objects are used to transport BaseAdvertisementObject objects defined in [RFC8008]. This document specifies how to transport such BaseAdvertisementObject objects via the ALTO Protocol with the ALTO CDNI Advertisement Service. Similar to other ALTO services, this document defines the ALTO information resource for the CDNI Advertisement Service as follows.

Note that the encoding of BaseAdvertisementObject reuses the one defined in [RFC8008] and therefore also follows the recommendations of I-JSON (Internet JSON) [RFC7493], which is required by [RFC8008].

3.1. Media Type

The media type of the CDNI Advertisement resource is "application/alto-cdni+json" (see Section 7).

3.2. HTTP Method

A CDNI Advertisement resource is requested using the HTTP GET method.

3.3. Accept Input Parameters

There are no applicable Accept Input parameters.

3.4. Capabilities

There are no applicable capabilities.

3.5. Uses

The "uses" field **MUST NOT** appear unless the CDNI Advertisement resource depends on other ALTO information resources. If the CDNI Advertisement resource has dependent resources, the resource IDs of its dependent resources **MUST** be included into the "uses" field. This document only defines one potential dependent resource for the CDNI Advertisement resource. See Section 4 for details of when and how to use it. Future documents may extend the CDNI Advertisement resource and allow other dependent resources.

3.6. Response

The "meta" field of a CDNI Advertisement response **MUST** include the "vtag" field defined in Section 10.3 of [RFC7285]. This field provides the version of the retrieved CDNI FCI resource.

If a CDNI Advertisement response depends on other ALTO information resources, it **MUST** include the "dependent-vtags" field, whose value is an array to indicate the version tags of the resources used, where each resource is specified in "uses" of its Information Resource Directory (IRD) entry.

The data component of an ALTO CDNI Advertisement response is named "cdni-advertisement", which is a JSON object of type CDNIAdvertisementData:

```
object {
  CDNIAdvertisementData cdni-advertisement;
} InfoResourceCDNIAdvertisement : ResponseEntityBase;

object {
  BaseAdvertisementObject capabilities-with-footprints<0..*>;
} CDNIAdvertisementData;
```

Specifically, a CDNIAdvertisementData object is a JSON object that includes only one property named "capabilities-with-footprints", whose value is an array of BaseAdvertisementObject objects. It provides capabilities with footprint restrictions for the uCDN to decide the dCDN selection. If the value of this property is an empty array, it means the corresponding dCDN cannot provide any mandatory-to-implement CDNI capabilities for any footprints.

The syntax and semantics of BaseAdvertisementObject are well defined in Section 5.1 of [RFC8008]. A BaseAdvertisementObject object includes multiple properties, including "capability-type", "capability-value", and "footprints", where "footprints" are defined in Section 4.2.2.2 of [RFC8006].

An equivalent specification in the ALTO-style notation (see Section 8.2 of [RFC7285]) creates a self-contained description of the BaseAdvertisementObject. As mentioned above, the normative specification of BaseAdvertisementObject is in [RFC8008].

```
object {
  JSONString capability-type;
  JSONValue capability-value;
  Footprint footprints<0..*>;
} BaseAdvertisementObject;

object {
  JSONString footprint-type;
  JSONString footprint-value<1..*>;
} Footprint;
```

For each BaseAdvertisementObject, the ALTO client **MUST** interpret "footprints" appearing multiple times as if they appeared only once. If "footprints" in a BaseAdvertisementObject is null or empty or does not appear, the ALTO client **MUST** understand that the capabilities in this BaseAdvertisementObject have the "global" coverage, i.e., the corresponding dCDN can provide them for any Request Routing source.

Note: Further optimization of BaseAdvertisementObjects to effectively provide the advertisement of capabilities with footprint restrictions is certainly possible. For example, these two examples below both describe that the dCDN can provide capabilities ["http/1.1", "https/1.1"] for the same footprints. However, the latter one is smaller in its size.

```
EXAMPLE 1
    {
      "meta": {...},
      "cdni-advertisement": {
        "capabilities-with-footprints": [
          {
            "capability-type": "FCI.DeliveryProtocol",
            "capability-value": {
              "delivery-protocols": [
                "http/1.1"
              ]
            },
            "footprints": [
              <Footprint objects>
            ]
          },
          {

            "capability-type": "FCI.DeliveryProtocol",
            "capability-value": {
              "delivery-protocols": [
                "https/1.1"
              ]
            },
            "footprints": [
              <Footprint objects>
            ]
          }
        ]
      }
    }
```

```
EXAMPLE 2
    {
      "meta": {...},
      "cdni-advertisement": {
        "capabilities-with-footprints": [
          {
            "capability-type": "FCI.DeliveryProtocol",
            "capability-value": {
              "delivery-protocols": [
                "https/1.1",
                "http/1.1"
              ]
            },
            "footprints": [
              <Footprint objects>
            ]
          }
        ]
      }
    }
```

Since such optimizations are not required for the basic interconnection of CDNs, the specifics of such mechanisms are outside the scope of this document.

This document only requires the ALTO server to provide the initial FCI-specific CDNI Payload Types defined in [RFC8008] as the mandatory-to-implement CDNI capabilities.

3.7. Examples

3.7.1. IRD

Below is the IRD of a simple, example ALTO server. The server provides both base ALTO information resources (e.g., network maps) and CDNI FCI-related information resources (e.g., CDNI Advertisement resources), demonstrating a single, integrated environment.

Specifically, the IRD announces nine information resources as follows:

- two network maps,
- one CDNI Advertisement resource without dependency,
- one CDNI Advertisement resource depending on a network map,
- one filtered CDNI Advertisement resource to be defined in Section 5,
- one property map including "cdni-capabilities" as its entity property,
- one filtered property map including "cdni-capabilities" and "pid" as its entity properties, and
- two update stream services:
  - one for updating CDNI Advertisement resources,
  - one for updating property maps

```
GET /directory HTTP/1.1
Host: alto.example.com
Accept: application/alto-directory+json,application/alto-error+json

HTTP/1.1 200 OK
Content-Length: 3531
Content-Type: application/alto-directory+json

{
  "meta": {
    "default-alto-network-map": "my-default-network-map"
  },
  "resources": {
    "my-default-network-map": {
      "uri": "https://alto.example.com/networkmap",
      "media-type": "application/alto-networkmap+json"
    },
    "my-eu-netmap": {
      "uri": "https://alto.example.com/myeunetmap",
      "media-type": "application/alto-networkmap+json"
    },
    "my-default-cdnifci": {
      "uri": "https://alto.example.com/cdnifci",
      "media-type": "application/alto-cdni+json"
    },
    "my-cdnifci-with-pid-footprints": {
      "uri": "https://alto.example.com/networkcdnifci",
      "media-type": "application/alto-cdni+json",
      "uses": [ "my-eu-netmap" ]
    },
    "my-filtered-cdnifci": {
      "uri": "https://alto.example.com/cdnifci/filtered",
      "media-type": "application/alto-cdni+json",
      "accepts": "application/alto-cdnifilter+json"
    },
    "cdnifci-property-map": {
      "uri": "https://alto.example.com/propmap/full/cdnifci",
      "media-type": "application/alto-propmap+json",
      "uses": [ "my-default-cdni" ],
      "capabilities": {
        "mappings": {
          "ipv4": [ "my-default-cdni.cdni-capabilities" ],
          "ipv6": [ "my-default-cdni.cdni-capabilities" ],
          "countrycode": [
            "my-default-cdni.cdni-capabilities" ],
          "asn": [ "my-default-cdni.cdni-capabilities" ]
        }
      }
    },
    "filtered-cdnifci-property-map": {
      "uri": "https://alto.example.com/propmap/lookup/cdnifci-pid",
      "media-type": "application/alto-propmap+json",
      "accepts": "application/alto-propmapparams+json",
      "uses": [ "my-default-cdni", "my-default-network-map" ],
      "capabilities": {
        "mappings": {
          "ipv4": [ "my-default-cdni.cdni-capabilities",
```

```
                    "my-default-network-map.pid" ],
          "ipv6": [ "my-default-cdni.cdni-capabilities",
                    "my-default-network-map.pid" ],
          "countrycode": [
            "my-default-cdni.cdni-capabilities" ],
          "asn": [ "my-default-cdni.cdni-capabilities" ]
        }
      }
    },
    "update-my-cdni-fci": {
      "uri": "https://alto.example.com/updates/cdnifci",
      "media-type": "text/event-stream",
      "accepts": "application/alto-updatestreamparams+json",
      "uses": [
        "my-default-network-map",
        "my-eu-netmap",
        "my-default-cdnifci",
        "my-filtered-cdnifci",
        "my-cdnifci-with-pid-footprints"
      ],
      "capabilities": {
        "incremental-change-media-types": {
         "my-default-network-map": "application/json-patch+json",
         "my-eu-netmap": "application/json-patch+json",
         "my-default-cdnifci":
         "application/merge-patch+json,application/json-patch+json",
         "my-filtered-cdnifci":
         "application/merge-patch+json,application/json-patch+json",
         "my-cdnifci-with-pid-footprints":
         "application/merge-patch+json,application/json-patch+json"
        }
      }
    },
    "update-my-props": {
      "uri": "https://alto.example.com/updates/properties",
      "media-type": "text/event-stream",
      "uses": [
        "cdnifci-property-map",
        "filtered-cdnifci-property-map"
      ],
      "capabilities": {
        "incremental-change-media-types": {
         "cdnifci-property-map":
         "application/merge-patch+json,application/json-patch+json",
         "filtered-cdnifci-property-map":
         "application/merge-patch+json,application/json-patch+json"
        }
      }
    }
  }
}
```

3.7.2. A Basic Example

This basic example demonstrates a simple CDNI Advertisement resource, which does not depend on other resources. There are three BaseAdvertisementObjects in this resource and these objects' capabilities are "http/1.1" delivery protocol, ["http/1.1", "https/1.1"] delivery protocol, and "https/1.1" acquisition protocol, respectively.

```
GET /cdnifci HTTP/1.1
Host: alto.example.com
Accept: application/alto-cdni+json,application/alto-error+json

HTTP/1.1 200 OK
Content-Length: 1411
Content-Type: application/alto-cdni+json

{
  "meta": {
    "vtag": {
      "resource-id": "my-default-cdnifci",
      "tag": "da65eca2eb7a10ce8b059740b0b2e3f8eb1d4785"
    }
  },
  "cdni-advertisement": {
    "capabilities-with-footprints": [
      {
        "capability-type": "FCI.DeliveryProtocol",
        "capability-value": {
          "delivery-protocols": [
            "http/1.1"
          ]
        },
        "footprints": [
          {
            "footprint-type": "ipv4cidr",
            "footprint-value": [ "192.0.2.0/24" ]
          },
          {
            "footprint-type": "ipv6cidr",
            "footprint-value": [ "2001:db8::/32" ]
          }
        ]
      },
      {
        "capability-type": "FCI.DeliveryProtocol",
        "capability-value": {
          "delivery-protocols": [
            "https/1.1",
            "http/1.1"
          ]
        },
        "footprints": [
          {
            "footprint-type": "ipv4cidr",
            "footprint-value": [ "198.51.100.0/24" ]
          }
        ]
      },
      {
        "capability-type": "FCI.AcquisitionProtocol",
        "capability-value": {
          "acquisition-protocols": [
            "https/1.1"
          ]
        },
```

```
          "footprints": [
            {
              "footprint-type": "ipv4cidr",
              "footprint-value": [ "203.0.113.0/24" ]
            }
          ]
        }
      ]
    }
  }
```

3.7.3. Incremental Updates

A benefit of using ALTO to provide CDNI Advertisement resources is that such resources can be updated using ALTO incremental updates [RFC8895]. Below is an example that also shows the benefit of having both JSON merge patch and JSON patch to encode updates.

At first, an ALTO client requests updates for "my-default-cdnifci", and the ALTO server returns the "control-uri" followed by the full CDNI Advertisement response. Then when there is a change in the "delivery-protocols" in that "http/1.1" is removed (from ["http/1.1", "https/1.1"] to only "https/1.1") due to maintenance of the "http/1.1" clusters, the ALTO server regenerates the new CDNI Advertisement resource and pushes the full replacement to the ALTO client. Later on, the ALTO server notifies the ALTO client that "192.0.2.0/24" is added into the "ipv4" footprint object for delivery protocol "https/1.1" by sending the change encoded by JSON patch to the ALTO client.

```
POST /updates/cdnifci HTTP/1.1
Host: alto.example.com
Accept: text/event-stream,application/alto-error+json
Content-Type: application/alto-updatestreamparams+json
Content-Length: 94

{
  "add": {
    "my-cdnifci-stream": {
      "resource-id": "my-default-cdnifci"
    }
  }
}

HTTP/1.1 200 OK
Connection: keep-alive
Content-Type: text/event-stream

event: application/alto-updatestreamcontrol+json
data: {"control-uri":
data: "https://alto.example.com/updates/streams/3141592653589"}

event: application/alto-cdni+json,my-cdnifci-stream
data: { ... full CDNI Advertisement resource ... }

event: application/alto-cdni+json,my-cdnifci-stream
data: {
data:   "meta": {
data:     "vtag": {
data:       "tag": "dasdfa10ce8b059740bddsfasd8eb1d47853716"
data:     }
data:   },
data:   "cdni-advertisement": {
data:     "capabilities-with-footprints": [
data:       {
data:         "capability-type": "FCI.DeliveryProtocol",
data:         "capability-value": {
data:           "delivery-protocols": [
data:             "https/1.1"
data:           ]
data:         },
data:         "footprints": [
data:           { "footprint-type": "ipv4cidr",
data:             "footprint-value": [ "203.0.113.0/24" ]
data:           }
data:         ]
data:       },
data:       { ... other CDNI advertisement object ... }
data:     ]
data:   }
data: }

event: application/json-patch+json,my-cdnifci-stream
data: [
data:   { "op": "replace",
data:     "path": "/meta/vtag/tag",
data:     "value": "a10ce8b059740b0b2e3f8eb1d4785acd42231bfe"
```

```
data:   },
data:   { "op": "add",
data:     "path": "/cdni-advertisement/capabilities-with-footprints
/0/footprints/0/footprint-value/-",
data:     "value": "192.0.2.0/24"
data:   }
data: ]
```

4. CDNI Advertisement Service Using ALTO Network Map

4.1. Network Map Footprint Type: altopid

The ALTO Protocol defines a concept called Provider-defined Identifier (PID) to represent a group of IPv4 or IPv6 addresses to which can be applied the same management policy. The PID is an alternative to the predefined CDNI footprint types (i.e., "ipv4cidr", "ipv6cidr", "asn", and "countrycode").

To leverage this concept, this document defines a new CDNI Footprint Type called "altopid". A CDNI Advertisement resource can depend on an ALTO network map resource and use "altopid" footprints to compress its CDNI Footprint Payload.

Specifically, the "altopid" footprint type indicates that the corresponding footprint value is a list of PIDNames as defined in [RFC7285]. These PIDNames are references of PIDs in a network map resource. Hence a CDNI Advertisement resource using "altopid" footprints depends on a network map. For such a CDNI Advertisement resource, the resource ID of its dependent network map **MUST** be included in the "uses" field of its IRD entry, and the "dependent-vtags" field with a reference to this network map **MUST** be included in its response (see the example in Section 4.2.2).

4.2. Examples

The following examples use the same IRD given in Section 3.7.1.

4.2.1. ALTO Network Map for CDNI Advertisements

Below provides a sample network map whose resource ID is "my-eu-netmap". This map is referenced by the CDNI Advertisement example in Section 4.2.2.

```
GET /myeunetmap HTTP/1.1
Host: alto.example.com
Accept: application/alto-networkmap+json,application/alto-error+json

HTTP/1.1 200 OK
Content-Length: 344
Content-Type: application/alto-networkmap+json

{
  "meta": {
    "vtag": [
      { "resource-id": "my-eu-netmap",
        "tag": "3ee2cb7e8d63d9fab71b9b34cbf764436315542e"
      }
    ]
  },
  "network-map": {
    "south-france" : {
      "ipv4": [ "192.0.2.0/24", "198.51.100.0/25" ],
      "ipv6": [ "2001:db8::/32" ]
    },
    "germany": {
      "ipv4": [ "203.0.113.0/24" ]
    }
  }
}
```

4.2.2. ALTO PID Footprints in CDNI Advertisements

This example shows a CDNI Advertisement resource that depends on a network map described in Section 4.2.1.

```
GET /networkcdnifci HTTP/1.1
Host: alto.example.com
Accept: application/alto-cdni+json,application/alto-error+json

HTTP/1.1 200 OK
Content-Length: 736
Content-Type: application/alto-cdni+json

{
  "meta": {
    "dependent-vtags": [
      {
        "resource-id": "my-eu-netmap",
        "tag": "3ee2cb7e8d63d9fab71b9b34cbf764436315542e"
      }
    ]
  },
  "cdni-advertisement": {
    "capabilities-with-footprints": [
      { "capability-type": "FCI.DeliveryProtocol",
        "capability-value": [ "https/1.1" ],
        "footprints": [
          { "footprint-type": "altopid",
            "footprint-value": [ "south-france" ]
          }
        ]
      },
      { "capability-type": "FCI.AcquisitionProtocol",
        "capability-value": [ "https/1.1" ],
        "footprints": [
          { "footprint-type": "altopid",
            "footprint-value": [ "germany", "south-france" ]
          }
        ]
      }
    ]
  }
}
```

4.2.3. Incremental Updates

In this example, the ALTO client is interested in changes of "my-cdnifci-with-pid-footprints" and its dependent network map "my-eu-netmap". Considering two changes, the first one is to change footprints of the "https/1.1" delivery protocol capability, and the second one is to remove the "south-france" PID from the footprints of the "https/1.1" acquisition protocol capability.

```
POST /updates/cdnifci HTTP/1.1
Host: alto.example.com
Accept: text/event-stream,application/alto-error+json
Content-Type: application/alto-updatestreamparams+json
Content-Length: 185

{
  "add": {
    "my-eu-netmap-stream": {
      "resource-id": "my-eu-netmap"
    },
    "my-netmap-cdnifci-stream": {
      "resource-id": "my-cdnifci-with-pid-footprints"
    }
  }
}

HTTP/1.1 200 OK
Connection: keep-alive
Content-Type: text/event-stream

event: application/alto-updatestreamcontrol+json
data: {"control-uri":
data: "https://alto.example.com/updates/streams/3141592653590"}

event: application/alto-networkmap+json,my-eu-netmap-stream
data: { ... full Network Map of my-eu-netmap ... }

event: application/alto-cdnifci+json,my-netmap-cdnifci-stream
data: { ... full CDNI Advertisement resource ... }

event: application/json-patch+json,my-netmap-cdnifci-stream
data: [
data:   { "op": "replace",
data:     "path": "/meta/vtag/tag",
data:     "value": "dasdfa10ce8b059740bddsfasd8eb1d47853716"
data:   },
data:   { "op": "add",
data:     "path":
data:     "/cdni-advertisement/capabilities-with-footprints
/0/footprints/0/footprint-value/-",
data:     "value": "germany"
data:   }
data: ]

event: application/json-patch+json,my-netmap-cdnifci-stream
data: [
data:   { "op": "replace",
data:     "path": "/meta/vtag/tag",
data:     "value": "a10ce8b059740b0b2e3f8eb1d4785acd42231bfe"
data:   },
data:   { "op": "remove",
data:     "path":
data:     "/cdni-advertisement/capabilities-with-footprints
/1/footprints/0/footprint-value/1"
data:   }
data: ]
```

5. Filtered CDNI Advertisement Using CDNI Capabilities

Sections 3 and 4 describe the CDNI Advertisement Service that can be used to enable a uCDN to get capabilities with footprint restrictions from dCDNs. However, since always getting full CDNI Advertisement resources from dCDNs is inefficient, this document introduces a new service named "Filtered CDNI Advertisement Service" to allow a client to filter a CDNI Advertisement resource using a client-given set of CDNI capabilities. For each entry of the CDNI Advertisement response, an entry will only be returned to the client if it contains at least one of the client-given CDNI capabilities. The relationship between a filtered CDNI Advertisement resource and a CDNI Advertisement resource is similar to the relationship between a filtered network/cost map and a network/cost map.

5.1. Media Type

A filtered CDNI Advertisement resource uses the same media type defined for the CDNI Advertisement resource in Section 3.1: "application/alto-cdni+json".

5.2. HTTP Method

A filtered CDNI Advertisement resource is requested using the HTTP POST method.

5.3. Accept Input Parameters

The input parameters for a filtered CDNI Advertisement resource are supplied in the entity body of the POST request. This document specifies the input parameters with a data format indicated by the media type "application/alto-cdnifilter+json", which is a JSON object of type ReqFilteredCDNIAdvertisement where:

```
object {
    JSONString capability-type;
    JSONValue capability-value;
} CDNICapability;

object {
    CDNICapability cdni-capabilities<0..*>;
} ReqFilteredCDNIAdvertisement;
```

with fields:

capability-type: The same as Base Advertisement Object's "capability-type" defined in Section 5.1 of [RFC8008].

capability-value: The same as Base Advertisement Object's "capability-value" defined in Section 5.1 of [RFC8008].

cdni-capabilities: A list of CDNI capabilities defined in Section 5.1 of [RFC8008] for which footprints are to be returned. If this list is empty, the ALTO server **MUST** interpret it as a request for the full CDNI Advertisement resource. The ALTO server **MUST** interpret entries appearing in this list multiple times as if they appeared only once. If the ALTO server does not define any footprints for a CDNI capability, it **MUST** omit this capability from the response.

5.4. Capabilities

There are no applicable capabilities.

5.5. Uses

The same rules as for the "uses" field of the CDNI Advertisement resource apply (see Section 3.5).

5.6. Response

If the request is invalid, the response **MUST** indicate an error using ALTO Protocol error handling specified in Section 8.5 of [RFC7285].

Specifically, a filtered CDNI Advertisement request is invalid if:

- the value of "capability-type" is null;
- the value of "capability-value" is null; or
- the value of "capability-value" is inconsistent with "capability-type".

When a request is invalid, the ALTO server **MUST** return an "E_INVALID_FIELD_VALUE" error defined in Section 8.5.2 of [RFC7285], and the "value" field of the error message **SHOULD** indicate this CDNI capability.

The ALTO server returns a filtered CDNI Advertisement resource for a valid request. The format of a filtered CDNI Advertisement resource is the same as a full CDNI Advertisement resource (see Section 3.6).

The returned filtered CDNI Advertisement resource **MUST** contain all the BaseAdvertisementObject objects satisfying the following condition: the CDNI capability object of each included BaseAdvertisementObject object **MUST** follow two constraints:

- The "cdni-capabilities" field of the input includes a CDNI capability object X having the same "capability-type" as it.
- All the mandatory properties in its "capability-value" is a superset of mandatory properties in "capability-value" of X semantically.

See Section 5.7.1 for a concrete example.

The version tag included in the "vtag" field of the response **MUST** correspond to the full CDNI Advertisement resource from which the filtered CDNI Advertisement resource is provided. This ensures that a single, canonical version tag is used independently of any filtering that is requested by an ALTO client.

5.7. Examples

The following examples use the same IRD example as in Section 3.7.1.

5.7.1. A Basic Example

This example filters the full CDNI Advertisement resource in Section 3.7.2 by selecting only the "http/1.1" delivery protocol capability. Only the second BaseAdvertisementObject in the full resource will be returned because the second object's capability is "http/1.1" and "https/1.1" delivery protocols, which is the superset of "https/1.1" delivery protocol.

```
POST /cdnifci/filtered HTTP/1.1
Host: alto.example.com
Accept: application/alto-cdni+json
Content-Type: application/cdnifilter+json
Content-Length: 176

{
  "cdni-capabilities": [
    {
      "capability-type": "FCI.DeliveryProtocol",
      "capability-value": {
        "delivery-protocols": [ "https/1.1" ]
      }
    }
  ]
}

HTTP/1.1 200 OK
Content-Length: 570
Content-Type: application/alto-cdni+json

{
  "meta": {
    "vtag": {
      "resource-id": "my-filtered-cdnifci",
      "tag": "da65eca2eb7a10ce8b059740b0b2e3f8eb1d4785"
    }
  },
  "cdni-advertisement": {
    "capabilities-with-footprints": [
      {
        "capability-type": "FCI.DeliveryProtocol",
        "capability-value": {
          "delivery-protocols": [
            "https/1.1",
            "http/1.1"
          ]
        },
        "footprints": [
          {
            "footprint-type": "ipv4cidr",
            "footprint-value": [ "198.51.100.0/24" ]
          }
        ]
      }
    ]
  }
}
```

5.7.2. Incremental Updates

In this example, the ALTO client only cares about the updates of one advertisement object for delivery protocol capability whose value includes "https/1.1". Thus, it adds its limitation of capabilities in "input" field of the POST request.

```
POST /updates/cdnifci HTTP/1.1
Host: fcialtoupdate.example.com
Accept: text/event-stream,application/alto-error+json
Content-Type: application/alto-updatestreamparams+json
Content-Length: 346

{
  "add": {
    "my-filtered-fci-stream": {
      "resource-id": "my-filtered-cdnifci",
      "input": {
        "cdni-capabilities": [
          {
            "capability-type": "FCI.DeliveryProtocol",
            "capability-value": {
              "delivery-protocols": [ "https/1.1" ]
            }
          }
        ]
      }
    }
  }
}

HTTP/1.1 200 OK
Connection: keep-alive
Content-Type: text/event-stream

event: application/alto-updatestreamcontrol+json
data: {"control-uri":
data: "https://alto.example.com/updates/streams/3141592653590"}

event: application/alto-cdni+json,my-filtered-fci-stream
data: { ... filtered CDNI Advertisement resource ... }

event: application/json-patch+json,my-filtered-fci-stream
data: [
data:   {
data:     "op": "replace",
data:     "path": "/meta/vtag/tag",
data:     "value": "a10ce8b059740b0b2e3f8eb1d4785acd42231bfe"
data:   },
data:   { "op": "add",
data:     "path":
data:     "/cdni-advertisement/capabilities-with-footprints
/0/footprints/0/footprint-value/-",
data:     "value": "192.0.2.0/24"
data:   }
data: ]
```

6. Query Footprint Properties Using ALTO Property Map Service

Besides the requirement of retrieving footprints of given capabilities, another common requirement for uCDN is to query CDNI capabilities of given footprints.

Considering each footprint as an entity with properties including CDNI capabilities, a natural way to satisfy this requirement is to use the ALTO property map as defined in [RFC9240]. This section describes how ALTO clients look up properties for individual footprints. First, it describes how to represent footprint objects as entities in the ALTO property map. Then it describes how to represent footprint capabilities as entity properties in the ALTO property map. Finally, it provides examples of the full property map and the filtered property map supporting CDNI capabilities, and their incremental updates.

6.1. Representing Footprint Objects as Property Map Entities

A footprint object has two properties: "footprint-type" and "footprint-value". A "footprint-value" is an array of footprint values conforming to the specification associated with the registered footprint type ("ipv4cidr", "ipv6cidr", "asn", "countrycode", and "altopid"). Considering each ALTO entity defined in [RFC9240] also has two properties: entity domain type and domain-specific identifier, a straightforward approach to represent a footprint as an ALTO entity is to represent its "footprint-type" as an entity domain type, and its footprint value as a domain-specific identifier.

Each existing footprint type can be represented as an entity domain type as follows:

- According to [RFC9240], "ipv4" and "ipv6" are two predefined entity domain types, which can be used to represent "ipv4cidr" and "ipv6cidr" footprints respectively. Note that both "ipv4" and "ipv6" domains can include not only hierarchical addresses but also individual addresses. Therefore, a "ipv4cidr" or "ipv6cidr" footprint with the longest prefix can also be represented by an individual address entity. When the uCDN receives a property map with individual addresses in an "ipv4" or "ipv6" domain, it can translate them as corresponding "ipv4cidr" or "ipv6cidr" footprints with the longest prefix.
- "pid" is also a predefined entity domain type, which can be used to represent "altopid" footprints. Note that "pid" is a resource-specific entity domain. To represent an "altopid" footprint, the specifying information resource of the corresponding "pid" entity domain **MUST** be the dependent network map used by the CDNI Advertisement resource providing this "altopid" footprint.
- However, no existing entity domain type can represent "asn" and "countrycode" footprints. To represent footprint-type "asn" and "countrycode", this document registers two new entity domains in Section 7 in addition to the ones in [RFC9240].

Here is an example of representing a footprint object of "ipv4cidr" type as a set of "ipv4" entities in the ALTO property map. The representation of the footprint object of "ipv6cidr" type is similar.

```
{ "footprint-type": "ipv4cidr",
  "footprint-value": ["192.0.2.0/24", "198.51.100.0/24"]
} --> "ipv4:192.0.2.0/24", "ipv4:198.51.100.0/24"
```

And here is an example of the corresponding footprint object of "ipv4cidr" type represented by an individual address in an "ipv4" domain in the ALTO property map. The translation of the entities in an "ipv6" domain is similar.

```
"ipv4:203.0.113.100" --> {
  "footprint-type": "ipv4cidr",
  "footprint-value": ["203.0.113.100/32"]
}
```

6.1.1. ASN Domain

The ASN domain associates property values with Autonomous Systems in the Internet.

6.1.1.1. Entity Domain Type

The entity domain type of the ASN domain is "asn" (in lowercase).

6.1.1.2. Domain-Specific Entity Identifiers

The entity identifier of an entity in an ASN domain **MUST** be encoded as a string consisting of the characters "as" (in lowercase) followed by the ASN [RFC6793] as a decimal number without leading zeros.

6.1.1.3. Hierarchy and Inheritance

There is no hierarchy or inheritance for properties associated with ASN.

6.1.2. COUNTRYCODE Domain

The COUNTRYCODE domain associates property values with countries.

6.1.2.1. Entity Domain Type

The entity domain type of the COUNTRYCODE domain is "countrycode" (in lowercase).

6.1.2.2. Domain-Specific Entity Identifiers

The entity identifier of an entity in a COUNTRYCODE domain is encoded as an ISO 3166-1 alpha-2 code [ISO3166-1] in lowercase.

6.1.2.3. Hierarchy and Inheritance

There is no hierarchy or inheritance for properties associated with country codes.

6.2. Representing CDNI Capabilities as Property Map Entity Properties

This document defines a new entity property type called "cdni-capabilities". An ALTO server can provide a property map resource mapping the "cdni-capabilities" entity property type for a CDNI Advertisement resource that it provides to an "ipv4", "ipv6", "asn", or "countrycode" entity domain.

6.2.1. Defining Information Resource Media Type for Property Type cdni-capabilities

The entity property type "cdni-capabilities" allows defining resource-specific entity properties. When resource-specific entity properties are defined with entity property type "cdni-capabilities", the defining information resource for a "cdni-capabilities" property **MUST** be a CDNI Advertisement resource provided by the ALTO server. The media type of the defining information resource for a "cdni-capabilities" property is therefore:

application/alto-cdni+json

6.2.2. Intended Semantics of Property Type cdni-capabilities

The purpose of a "cdni-capabilities" property for an entity is to indicate all the CDNI capabilities that a corresponding CDNI Advertisement resource provides for the footprint represented by this entity. Thus, the value of a "cdni-capabilities" property **MUST** be a JSON array. Each element in a "cdni-capabilities" property **MUST** be a JSON object in the format of CDNICapability (see Section 5.3). The value of a "cdni-capabilities" property for an "ipv4", "ipv6", "asn", "countrycode", or "altopid" entity **MUST** include all the CDNICapability objects satisfying the following conditions: (1) they are provided by the defining CDNI Advertisement resource, and (2) the represented footprint object of this entity is in their footprint restrictions.

6.3. Examples

The following examples use the same IRD example given by Section 3.7.1.

6.3.1. Property Map

This example shows a full property map in which entities are footprints and entities' property is "cdni-capabilities".

```
GET /propmap/full/cdnifci HTTP/1.1
Host: alto.example.com
Accept: application/alto-propmap+json,application/alto-error+json

HTTP/1.1 200 OK
Content-Length: 1522
Content-Type: application/alto-propmap+json

{
  "property-map": {
    "meta": {
      "dependent-vtags": [
        { "resource-id": "my-default-cdnifci",
          "tag": "7915dc0290c2705481c491a2b4ffbec482b3cf62"}
      ]
    },
    "countrycode:us": {
      "my-default-cdnifci.cdni-capabilities": [
        { "capability-type": "FCI.DeliveryProtocol",
          "capability-value": {
            "delivery-protocols": ["http/1.1"]}}]
    },
    "ipv4:192.0.2.0/24": {
      "my-default-cdnifci.cdni-capabilities": [
        { "capability-type": "FCI.DeliveryProtocol",
          "capability-value": {
            "delivery-protocols": ["http/1.1"]}}]
    },
    "ipv4:198.51.100.0/24": {
      "my-default-cdnifci.cdni-capabilities": [
        { "capability-type": "FCI.DeliveryProtocol",
          "capability-value": {
            "delivery-protocols": ["https/1.1", "http/1.1"]}}]
    },
    "ipv4:203.0.113.0/24": {
      "my-default-cdnifci.cdni-capabilities": [
        { "capability-type": "FCI.AcquisitionProtocol",
          "capability-value": {
            "acquisition-protocols": ["http/1.1"]}}]
    },
    "ipv6:2001:db8::/32": {
      "my-default-cdnifci.cdni-capabilities": [
        { "capability-type": "FCI.DeliveryProtocol",
          "capability-value": {
            "delivery-protocols": ["http/1.1"]}}]
    },
    "asn:as64496": {
      "my-default-cdnifci.cdni-capabilities": [
        { "capability-type": "FCI.DeliveryProtocol",
          "capability-value": {
            "delivery-protocols": ["https/1.1", "http/1.1"]}}]
    }
  }
}
```

6.3.2. Filtered Property Map

This example uses the filtered property Map Service to get "pid" and "cdni-capabilities" properties for two footprints "ipv4:192.0.2.0/24" and "ipv6:2001:db8::/32".

```
   POST /propmap/lookup/cdnifci-pid HTTP/1.1
   Host: alto.example.com
   Content-Type: application/alto-propmapparams+json
   Accept: application/alto-propmap+json,application/alto-error+json
   Content-Length: 181

   {
     "entities": [
       "ipv4:192.0.2.0/24",
       "ipv6:2001:db8::/32"
     ],
     "properties": [ "my-default-cdnifci.cdni-capabilities",
                     "my-default-networkmap.pid" ]
   }

 HTTP/1.1 200 OK
 Content-Length: 796
 Content-Type: application/alto-propmap+json

 {
   "property-map": {
     "meta": {
       "dependent-vtags": [
          {"resource-id": "my-default-cdnifci",
            "tag": "7915dc0290c2705481c491a2b4ffbec482b3cf62"},
          {"resource-id": "my-default-networkmap",
            "tag": "7915dc0290c2705481c491a2b4ffbec482b3cf63"}
       ]
     },
     "ipv4:192.0.2.0/24": {
       "my-default-cdnifci.cdni-capabilities": [
         {"capability-type": "FCI.DeliveryProtocol",
          "capability-value": {"delivery-protocols": ["http/1.1"]}}],
       "my-default-networkmap.pid": "pid1"
     },
     "ipv6:2001:db8::/32": {
       "my-default-cdnifci.cdni-capabilities": [
         {"capability-type": "FCI.DeliveryProtocol",
          "capability-value": {"delivery-protocols": ["http/1.1"]}}],
       "my-default-networkmap.pid": "pid3"
     }
   }
 }
```

6.3.3. Incremental Updates

In this example, the ALTO client is interested in updates for the properties "cdni-capabilities" and "pid" of two footprints "ipv4:192.0.2.0/24" and "countrycode:fr".

```
POST /updates/properties HTTP/1.1
Host: alto.example.com
Accept: text/event-stream,application/alto-error+json
Content-Type: application/alto-updatestreamparams+json
Content-Length: 339

{
  "add": {
    "fci-propmap-stream": {
      "resource-id": "filtered-cdnifci-property-map",
      "input": {
        "properties": [ "my-default-cdnifci.cdni-capabilities",
                        "my-default-networkmap.pid" ],
        "entities": [ "ipv4:192.0.2.0/24",
                      "ipv6:2001:db8::/32" ]
      }
    }
  }
}

HTTP/1.1 200 OK
Connection: keep-alive
Content-Type: text/event-stream

event: application/alto-updatestreamcontrol+json
data: {"control-uri":
data: "https://alto.example.com/updates/streams/1414213562373"}

event: application/alto-cdni+json,fci-propmap-stream
data: { ... filtered property map ... }

event: application/merge-patch+json,fci-propmap-stream
data: {
data:   "property-map": {
data:     "meta": {
data:       "dependent-vtags": [
data:         { "resource-id": "my-default-cdnifci",
data:           "tag": "2beeac8ee23c3dd1e98a73fd30df80ece9fa5627"},
data:         { "resource-id": "my-default-networkmap",
data:           "tag": "7915dc0290c2705481c491a2b4ffbec482b3cf63"}
data:       ]
data:     },
data:     "ipv4:192.0.2.0/24": {
data:       "my-default-cdnifci.cdni-capabilities": [
data:         { "capability-type": "FCI.DeliveryProtocol",
data:           "capability-value": {
data:             "delivery-protocols": ["http/1.1", "https/1.1"]}}]
data:     }
data:   }
data: }

event: application/json-patch+json,fci-propmap-stream
data: [
data:   { "op": "replace",
data:     "path": "/meta/dependent-vtags/0/tag",
data:     "value": "61b23185a50dc7b334577507e8f00ff8c3b409e4"
data:   },
```

```
data:   { "op": "replace",
data:     "path":
data:     "/property-map/countrycode:fr/my-default-networkmap.pid",
data:     "value": "pid5"
data:   }
data: ]
```

7. IANA Considerations

This document defines two new media types: "application/alto-cdni+json", as described in Section 7.1, and "application/cdnifilter+json", as described in Section 7.2. It also defines a new CDNI Metadata Footprint Type (Section 7.3), two new ALTO entity domain types (Section 7.4), and a new ALTO entity property type (Section 7.5).

7.1. application/alto-cdni+json Media Type

Type name:
application

Subtype name:
alto-cdni+json

Required parameters:
N/A

Optional parameters:
N/A

Encoding considerations:
Encoding considerations are identical to those specified for the "application/json" media type. See [RFC8259].

Security considerations:
Security considerations related to the generation and consumption of ALTO Protocol messages are discussed in Section 15 of [RFC7285].

Interoperability considerations:
N/A

Published specification:
Section 3 of RFC 9241

Applications that use this media type:
ALTO servers and ALTO clients [RFC7285] either stand alone or are embedded within other applications that provide CDNI interfaces for uCDNs or dCDNs.

Fragment identifier considerations:
N/A

Additional information:
Magic number(s): N/A

File extension(s): N/A

Macintosh file type code(s): N/A

Person & email address to contact for further information:
See Authors' Addresses section.

Intended usage:
COMMON

Restrictions on usage:
N/A

Author:
See Authors' Addresses section.

Change controller:
Internet Engineering Task Force (iesg@ietf.org)

7.2. application/alto-cdnifilter+json Media Type

Type name:
application

Subtype name:
alto-cdnifilter+json

Required parameters:
N/A

Optional parameters:
N/A

Encoding considerations:
Encoding considerations are identical to those specified for the "application/json" media type. See [RFC8259].

Security considerations:
Security considerations related to the generation and consumption of ALTO Protocol messages are discussed in Section 15 of [RFC7285].

Interoperability considerations:
N/A

Published specification:
Section 5 of RFC 9241

Applications that use this media type:
ALTO servers and ALTO clients [RFC7285] either stand alone or are embedded within other applications that provide CDNI interfaces for uCDNs or dCDNs and supports CDNI capability-based filtering.

Fragment identifier considerations:
N/A

Additional information:
Magic number(s): N/A

File extension(s): N/A

Macintosh file type code(s): N/A

Person & email address to contact for further information:
See Authors' Addresses section.

Intended usage:
COMMON

Restrictions on usage:
N/A

Author:
See Authors' Addresses section.

Change controller:
Internet Engineering Task Force (iesg@ietf.org)

7.3. CDNI Metadata Footprint Types Registry

This document updates the "CDNI Metadata Footprint Types" registry created by Section 7.2 of [RFC8006]. A new footprint type, which is listed in Table 1, has been registered.

Footprint Type	Description	Reference
altopid	A list of PID names	RFC 9241, Section 4

Table 1: CDNI Metadata Footprint Type

7.4. ALTO Entity Domain Types Registry

This document updates the "ALTO Entity Domain Types" registry created by Section 11.2 of [RFC9240]. Two new entity domain types, which are listed in Table 2, have been registered.

Identifier	Entity Identifier Encoding	Hierarchy and Inheritance	Media Type of Defining Resource	Mapping to ALTO Address Type
asn	See RFC 9241, Section 6.1.1.2	None	None	false
countrycode	See RFC 9241, Section 6.1.2.2	None	None	false

Table 2: Additional ALTO Entity Domain Types

7.5. ALTO Entity Property Types Registry

This document updates the "ALTO Entity Property Types" registry created by Section 11.3 of [RFC9240]. A new entity property type, which is listed in Table 3, has been registered.

Identifier	Intended Semantics	Media Type of Defining Resource
cdni-capabilities	See RFC 9241, Section 6.2	application/alto-cdni+json

Table 3: Additional ALTO Entity Property Type

8. Security Considerations

As an extension of the base ALTO Protocol [RFC7285], this document fits into the architecture of the base protocol, and hence Security Considerations of the base protocol (Section 15 of [RFC7285]) fully apply when this extension is provided by an ALTO server.

In the context of CDNI Advertisement, the following security risk scenarios should be considered:

- Authenticity and integrity of ALTO information: an attacker may disguise itself as an ALTO server for a dCDN (e.g., by starting a on-path attack) and provide false capabilities and footprints to a uCDN using the CDNI Advertisement Service. Such false information may lead a uCDN to (1) select an incorrect dCDN to serve user requests or (2) skip uCDNs in good conditions. To address this risk, protection strategies in Section 15.1.2 of [RFC7285] can be applied.
- Potential undesirable guidance from authenticated ALTO information: a dCDN can provide a uCDN with limited capabilities and smaller footprint coverage so that the dCDN can avoid transferring traffic for a uCDN that they should have to transfer. To reduce this risk, the protection strategies in Section 15.2.2 of [RFC7285] can be considered.
- Confidentiality and privacy of ALTO information: footprint properties integrated with ALTO property maps may expose network location identifiers (e.g., IP addresses or fine-grained PIDs). To address this risk, the protection strategy for risk types (1) and (3) as described in Section 15.3 of [RFC7285] can be considered.
- For availability of ALTO services, an attacker may conduct service-degradation attacks using services defined in this document to disable ALTO services of a network. It may request

potentially large, full CDNI Advertisement resources from an ALTO server in a dCDN continuously in order to consume the bandwidth resources of that ALTO server. It may also query filtered property Map Services with many smaller individual footprints in order to consume the computation resources of the ALTO server. To mitigate these risks, the protection strategies in Section 15.5.2 of [RFC7285] can be applied.

Although protection strategies as described in Section 15 of [RFC7285] should be applied to address aforementioned security and privacy considerations, two special cases need to be included as follows:

- As required by Section 7 of [RFC8008],

  > All protocols that implement these capabilities and footprint advertisement objects are **REQUIRED** to provide integrity and authentication services.

  Therefore, the uCDN (ALTO Client) **MUST** be authenticated to the dCDN (ALTO Server). And the dCDN (ALTO Server) **MUST** support HTTP Digest Authentication [RFC7616] and **MAY** also support TLS mutual authentication [RFC8446]. The authentication method will need to be negotiated out of band and is out of scope for this document, as is the approach for provisioning and managing these credentials.
- One specific information leakage risk introduced by this document cannot be addressed by these strategies. In particular, if a dCDN A signs agreements with multiple uCDNs without any isolation, dCDN A may disclose extra information of one uCDN to another one. In that case, one uCDN may redirect requests that should not have to be served by dCDN A to dCDN A.

  To reduce the risk, a dCDN **SHOULD** isolate full and/or filtered CDNI Advertisement resources for different uCDNs. It could consider generating URIs of different full and/or filtered CDNI Advertisement resources by hashing its company ID, a uCDN's company ID as well as their agreements. A dCDN **SHOULD** avoid exposing all full and/or filtered CDNI Advertisement resources in one of its IRDs.

9. References

9.1. Normative References

[ISO3166-1] International Organization for Standardization, "Codes for the representation of names of countries and their subdivisions -- Part 1: Country codes", ISO 3166-1:2020, August 2020.

[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/info/rfc2119>.

[RFC6793] Vohra, Q. and E. Chen, "BGP Support for Four-Octet Autonomous System (AS) Number Space", RFC 6793, DOI 10.17487/RFC6793, December 2012, <https://www.rfc-editor.org/info/rfc6793>.

[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/info/rfc7285>.

[RFC7493] Bray, T., Ed., "The I-JSON Message Format", RFC 7493, DOI 10.17487/RFC7493, March 2015, <https://www.rfc-editor.org/info/rfc7493>.

[RFC7616] Shekh-Yusef, R., Ed., Ahrens, D., and S. Bremer, "HTTP Digest Access Authentication", RFC 7616, DOI 10.17487/RFC7616, September 2015, <https://www.rfc-editor.org/info/rfc7616>.

[RFC8006] Niven-Jenkins, B., Murray, R., Caulfield, M., and K. Ma, "Content Delivery Network Interconnection (CDNI) Metadata", RFC 8006, DOI 10.17487/RFC8006, December 2016, <https://www.rfc-editor.org/info/rfc8006>.

[RFC8008] Seedorf, J., Peterson, J., Previdi, S., van Brandenburg, R., and K. Ma, "Content Delivery Network Interconnection (CDNI) Request Routing: Footprint and Capabilities Semantics", RFC 8008, DOI 10.17487/RFC8008, December 2016, <https://www.rfc-editor.org/info/rfc8008>.

[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/info/rfc8174>.

[RFC8259] Bray, T., Ed., "The JavaScript Object Notation (JSON) Data Interchange Format", STD 90, RFC 8259, DOI 10.17487/RFC8259, December 2017, <https://www.rfc-editor.org/info/rfc8259>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

[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/info/rfc8895>.

[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/info/rfc9240>.

9.2. Informative References

[ALTO-PATH-VECTOR] Gao, K., Lee, Y., Randriamasy, S., Yang, Y. R., and J. J. Zhang, "An ALTO Extension: Path Vector", Work in Progress, Internet-Draft, draft-ietf-alto-path-vector-25, 20 March 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-alto-path-vector-25>.

[RFC5693] Seedorf, J. and E. Burger, "Application-Layer Traffic Optimization (ALTO) Problem Statement", RFC 5693, DOI 10.17487/RFC5693, October 2009, <https://www.rfc-editor.org/info/rfc5693>.

[RFC6707] Niven-Jenkins, B., Le Faucheur, F., and N. Bitar, "Content Distribution Network Interconnection (CDNI) Problem Statement", RFC 6707, DOI 10.17487/RFC6707, September 2012, <https://www.rfc-editor.org/info/rfc6707>.

[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/info/rfc7971>.

[RFC7975] Niven-Jenkins, B., Ed. and R. van Brandenburg, Ed., "Request Routing Redirection Interface for Content Delivery Network (CDN) Interconnection", RFC 7975, DOI 10.17487/RFC7975, October 2016, <https://www.rfc-editor.org/info/rfc7975>.

Acknowledgments

The authors thank Matt Caulfield, Danny Alex Lachos Perez, Daryl Malas, and Sanjay Mishra for their timely reviews and invaluable comments. Big thanks also to the ALTO WG Chairs (Qin Wu and Vijay Gurbani), all the directorate reviewers, and the IESG reviewers (Martin Duke, Erik Kline, Martin Vigoureux, Murray Kucherawy, Roman Danyliw, Zaheduzzaman Sarker, Éric Vyncke, and Francesca Palombini), for their thorough reviews, discussions, guidance, and shepherding, which further improve this document.

Jan Seedorf has been partially supported by the GreenICN project (GreenICN: Architecture and Applications of Green Information Centric Networking), a research project supported jointly by the European Commission under its 7th Framework Program (contract no. 608518) and the National Institute of Information and Communications Technology (NICT) in Japan (contract no. 167). The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the GreenICN project, the European Commission, or NICT.

This document has also been supported by the Coordination Support Action entitled 'Supporting European Experts Presence in International Standardisation Activities in ICT' (StandICT.eu) funded by the European Commission under the Horizon 2020 Programme with Grant Agreement no. 780439. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the European Commission.

Contributors

Xiao Shawn Lin
Huawei
2222 Newjinqiao Rd
Shanghai
200125
China
Phone: +86-15316812351
Email: x.shawn.lin@gmail.com

Authors' Addresses

Jan Seedorf
HFT Stuttgart - Univ. of Applied Sciences
Schellingstrasse 24
70174 Stuttgart
Germany
Phone: +49-0711-8926-2801
Email: jan.seedorf@hft-stuttgart.de

Y. Richard Yang
Yale University
51 Prospect Street
New Haven, CT 06511
United States of America
Phone: +1-203-432-6400
Email: yry@cs.yale.edu
URI: http://www.cs.yale.edu/~yry/

Kevin J. Ma
Ericsson
43 Nagog Park
Acton, MA 01720
United States of America
Phone: +1-978-844-5100
Email: kevin.j.ma.ietf@gmail.com

Jon Peterson
NeuStar
1800 Sutter St., Suite 570
Concord, CA 94520
United States of America
Email: jon.peterson@neustar.biz

Jingxuan Jensen Zhang
Tongji University
4800 Cao'an Hwy
Shanghai
201804
China
Email: jingxuan.zhang@tongji.edu.cn