Workgroup: Network Working Group
Internet-Draft:
draft-ietf-cdni-capacity-insights-
extensions-01
Updates: 8006, 8008 (if approved)
Published: 13 March 2023
Intended Status: Standards Track
Expires: 14 September 2023
Authors: A. Ryan, Disney Streaming
B. Rosenblum, Vecima
N. Sopher, Qwilt

CDNI Capacity Capability Advertisement Extensions

Abstract

Open Caching architecture is a use case of Content Delivery Networks Interconnection (CDNI) in which the commercial Content Delivery Network (CDN) is the upstream CDN (uCDN) and the ISP caching layer serves as the downstream CDN (dCDN). This document supplements to the CDNI Capability Objects defined in RFC 8008 the defined capability objects structure and interface for advertisements and management of a downstream CDN capacity.

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 14 September 2023.

Table of Contents

1. Introduction

The Streaming Video Technology Alliance [SVTA] is a global association that works to solve streaming video challenges in an effort to improve end-user experience and adoption. The Open Caching Working Group [OCWG] of the Streaming Video Technology Alliance [SVTA] is focused on the delegation of video delivery requests from commercial CDNs to a caching layer at the ISP's network. Open Caching architecture is a specific use case of CDNI where the commercial CDN is the upstream CDN (uCDN) and the ISP caching layer is the downstream CDN (dCDN). While delegating traffic from one CDN to the other, it is important to make sure that an appropriate amount of traffic is delegated. In order to achieve that, the SVTA Open Caching Capacity Insight Specification [OC-CII] defines a feedback mechanism to inform the delegator how much traffic is appropriate to delegate. The traffic level information provided by that interface will be consumed by entities, such as the Open Caching Request router [OC-RR], to help inform that entity's traffic delegation decisions. This document defines and registers CDNI Payload Types (as defined at section 7.1 of [RFC8006]). These Payload types are used for Capability Objects added to those defined

at section 4 of [RFC8008], which are required for the Open Caching Capacity Insights Interface [OC-CII].

For consistency with other CDNI documents this document follows the CDNI convention of uCDN (upstream CDN) and dCDN (downstream CDN) to represent the commercial CDN and ISP caching layer respectively.

This document registers two CDNI Payload Types (section 7.1 of [RFC8006]) for the defined capability objects:

- Telemetry Payload Type: A payload type for the capability object which defines supported telemetry sources, the metrics made available by that source, and corresponding configuration appropriate to the type of the source (host, port, protocol, etc..).

- CapacityLimits Payload Type: a payload type for the capability object which defines Capacity Limits based on a set of defined limit types and a mapping from those limits to corresponding telemetry sources for supporting real-time metrics.

1.1. Terminology

The following terms are used throughout this document:

- CDN - Content Delivery Network

Additionally, this document reuses the terminology defined in [RFC6707], [RFC7336], [RFC8006], [RFC8007], [RFC8008], and [RFC8804]. Specifically, we use the following CDNI acronyms:

- uCDN, dCDN - Upstream CDN and Downstream CDN respectively (see [RFC7336])

1.2. 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.

1.3. Objectives

In order to enable information exchange between a uCDN and a dCDN about acceptable levels of traffic to delegate, the following process has been defined:

In normal operation a uCDN will communicate with a dCDN, via an interface, to collect and understand any limits that a dCDN has set forth for traffic delegation from a uCDN. These limits will come in the form of metrics such as bits per second, requests per second, etc.. These limits can be thought of as Not to Exceed (NTE) limits.

The dCDN should provide access to a telemetry source, of near real time metrics, that the uCDN can use to track current usage. The uCDN should compare its current usage to the limits the dCDN has put forth and adjust traffic delegation decisions accordingly to keep current usage under the specified limits.

In summary, the dCDN will provide the uCDN limits on how much traffic it should delegate towards the dCDN and then also provide a telemetry source that is coupled to the same scope as the limit, so that the uCDN can use it to track its current usage against the advertised limit. Having a limit and a corresponding telemetry source for that limit allows for a non ambiguous definition of what a particular limit means for both the uCDN and dCDN.

Limits that are communicated from the dCDN to the uCDN should be considered valid based on the TTL of the response. The TTL of the response will be provided by the transport mechanism for the response , i.e., an HTTP Cache-Control header. The intention is that the limits would have a long lived TTL and would represent a reasonable peak utilization limit that the uCDN should target.

In the event a dCDN needs to inform a uCDN of an update to a previously communicated limit, the dCDN SHOULD be able to leverage a uCDN callback endpoint to inform the uCDN of adjusted limits. The most common use case for this would be related to dCDN infrastructure issues which reduced the amount of capacity previously advertised as being available.

2. CDNI Additional Capability Objects

Section 5 of [RFC8008] describes the FCI Capability Advertisement Object, which contains a CDNI Capability Object as well as the capability object type (a CDNI Payload Type). The section also defines the Capability Objects per such type. Below we define two additional Capability Objects.

Note: In the following sections, the term "mandatory-to-specify" is used to convey which properties MUST be included when serializing a given capability object. When mandatory-to-specify is defined as "Yes" for an individual property, it means that if the object containing that property is included in an FCI message, then the mandatory-to-specify property MUST also be included.

2.1. Telemetry Capability Object

The Telemetry Capability Object is used to define a list of telemetry sources made available by the dCDN to the uCDN. In this document, Telemetry data is being defined as near real time aggregated metrics of dCDN utilization, such as bits per second egress, and should be specific to the uCDN and dCDN traffic delegation relationship. Telemetry data is uniquely defined by a source id, a metrics name, along with the footprints that are associated with an FCI.Capability advertisement. When defining a

Capacity Limit, the meaning of a limit might be considered ambiguous if the uCDN and dCDN are defining current usage via different data sources. Having the dCDN provide a data source defining usage that both itself and the uCDN reference allows a non ambiguous metric to use when determining current usage and how that compared to a limit. Telemetry data is not only an important component for making informed traffic delegation decisions but also for providing visibility to traffic that has been delegated back through to upstream providers. In situations where there are multiple CDNI delegations, a uCDN will need to incorporate the usage information from any dCDN's it delegated to when itself is asked to provide usage information otherwise, the traffic may seem unaccounted for. An example of this situation is when a Content Provider delegates traffic directly to a CDN, and that CDN decides to further delegate that traffic to a dCDN, if the Content Provider polls the uCDN for traffic usage, if the uCDN does not integrate the Telemetry data of the dCDN it delegated to, any of the traffic the uCDN delegated to its dCDN would become invisible to the Content Provider.

Property: sources

Description: Telemetry sources made available to the uCDN.

Type: A JSON array of Telemetry Source objects (see Section 2.1.1).

Mandatory-to-Specify: Yes.

2.1.1. Telemetry Source Object

The Telemetry Source Object is built of an associated type, a list of exposed metrics, and type-specific configuration data.

Property: id

Description: A unique identifier of a telemetry source.

Type: String.

Mandatory-to-Specify: Yes.

Property: type

Description: A valid telemetry source type. See Section 2.1.1.1.

Type: String.

Mandatory-to-Specify: Yes.

Property: metrics

Description: The metrics exposed by this source.

Type: A JSON array of Telemetry Source Metric objects (see Section 2.1.1.2).

Mandatory-to-Specify: Yes.

Property: configuration

Description: a source-specific representation of the Telemetry source configuration. For the generic source type, this configuration format is defined out-of-band. For other types, the configuration format will be specified in a yet to be defined Telemetry Interface specification. The goal of this element is to allow for forward compatibility with a formal Telemetry interface.

Type: A JSON object: TBD

Mandatory-to-Specify: No.

2.1.1.1. Telemetry Source Types

Below are the listed valid telemetry source types. At the time of this draft, the type registry is limited to a single type of Generic. The intention of this type registry is to allow for future extension to reference a yet to be drafted specification for a CDNI Telemetry interface, which would standardize the definition, format,etc of Telemetry data between participants of a CDNI workflow.

Source Type	Description
generic	An object which allows for advertisement of generic datasources

Table 1

2.1.1.2. Telemetry Source Metric Object

The Telemetry Source Metric Object describes the metric to be exposed.

Property: name

Description: An identifier unique within this telemetry source.

Type: String.

Mandatory-to-Specify: Yes.

Property: time-granularity

Description: Represents the time frame that the data represents in seconds. e.g., is this a data set over 300

seconds (i.e., 5 minutes), 3600 seconds (i.e., one hour), etc..

Type: Integer.

Mandatory-to-Specify: No.

Property: data-percentile

Description: The percentile calculation the data represents, i.e. 50 percentile would equate to the median over the time-granularity. Lack of a data-percentile will mean that the data is the average over the time representation.

Type: Integer.

Mandatory-to-Specify: No.

Property: latency

Description: Time in seconds that the data is behind of real time. This is important to specify to help the uCDN to understand how long it might take to reflect traffic adjustments in the metrics.

Type: Integer.

Mandatory-to-Specify: No.

2.1.2. Telemetry Capability Object Serialization

The following shows an example of Telemetry Capability including 2 metrics for a source, that is scoped to a footprint.

```
"capabilities": [
  {
    "capability-type": "FCI.Telemetry",
    "capability-value": {
      "sources": [
        {
          "id": "capacity_metrics_region1",
          "type": "generic",
          "metrics": [
            {
              "name": "egress_5m",
              "time-granularity": 300,
              "data-percentile": 50,
              "latency": 1500
            },
            {
              "name": "requests_5m",
               ...
            }
          ]
        }
      ]
    },
    "footprints": [
      <footprint objects>
    ]
  }
]
```

2.2. CapacityLimits Capability Object

The Capacity Limits Capability Object enables the dCDN to specify traffic delegation limits to a uCDN within an FCI.Capabilities advertisement. The limits specified by the dCDN will inform the uCDN on how much traffic can be delegated to the dCDN. The limits specified by the dCDN should be considered Not To Exceed (NTE) limits. The limits should be based on near real time telemetry data that the dCDN provides to the uCDN, or in other words, for each limit that is advertised, there should also exist a telemetry source which provides data of current utilization against the particular advertised limit.

Property: limits

Description: A collection of Capacity Limit objects.

Type: A JSON array of CapacityLimit objects (see Section 2.2.1).

Mandatory-to-Specify: Yes.

2.2.1. Capacity Limit Object

A CapacityLimit object is used to represent traffic limits for delegation from the uCDN towards the dCDN. The limit object is scoped to the footprint associated with the FCI capability advertisement encompassing this object. Limits will be considered using a logical AND, such that a uCDN will need to ensure that all the limits are considered and honored rather than choosing the most specific only.

Property: limit-type

Description: The units of maximum-hard and maximum-soft.

Type: String. One of the values listed in Section 2.2.1.1.

Mandatory-to-Specify: Yes.

Property: id

Description: Specifies a unique identifier associated with a limit. The is CAN be used as a relational identifier to a specific Section 2.2.1.

Type: String.

Mandatory-to-Specify: No.

Property: maximum-hard

Description: The maximum unit of capacity that is available for use.

Type: Integer.

Mandatory-to-Specify: Yes.

Property: maximum-soft

Description: A soft limit at which an upstream should consider deducing traffic to prevent hitting the hard limit.

Type: Integer.

Mandatory-to-Specify: No.

Property: current

Description: Specifies the current usage value of the limit. It is not recommended to specify the current usage value inline with the FCI.CapacityLimits advertisements as it will reduce the ability to cache the response. The intended method for providing telemetry data is to reference a Section 2.2.1.2 to poll for the current usage.

Type: Integer.

Mandatory-to-Specify: No.

Property: telemetry-source

Description: Mapping of each a particular limit to a specific metric with relevant real-time data provided by a telemetry source.

Type: Capacity Limit Telemetry Source object (see Section 2.2.1.2).

Mandatory-to-Specify: No.

2.2.1.1. Capacity Limit Types

Below are listed the valid capacity limit types. Additional limits would need to be specified and extended into this list. The values specified here represent the types that were identified as being the most relevant metrics for the purposes of traffic delegation between CDNs.

Limit Type	Units
egress	Bits per second
requests	Requests per second
storage-size	Total bytes
storage-objects	Count
sessions	Count
cache-size	Total bytes

Table 2

2.2.1.2. Capacity Limit Telemetry Source Object

The Capacity Limit Telemetry Source Object refers to a specific metric within a Telemetry Source.

Property: id

Description: Reference to the "id" of a telemetry source defined by a Telemetry Capability object.

Type: String.

Mandatory-to-Specify: Yes.

Property: metric

Description: Reference to the "name" property of a metric defined within a telemetry source of an FCI.Telemetry Capability object.

Type: String.

Mandatory-to-Specify: Yes.

2.2.2. Capacity Limit Object Serialization

The following shows an example of an FCI.CapacityLimits object.

```
"capabilities":[
  {
    "capability-type":"FCI.CapacityLimits",
    "capability-value":{
      "limits":[
        {
          "id":"capacity_limit_region1",
          "limit-type":"egress",
          "maximum-hard":50000000000,
          "maximum-soft":25000000000,
          "telemetry-source":{
            "id":"capacity_metrics_region1",
            "metric":"egress_5m"
          }
        }
      ]
    },
    "footprints":[
      "<footprint objects>"
    ]
  }
]
```

3. IANA Considerations

3.1. CDNI Payload Types

similar to the type definitions described in section 7.1 of [RFC8006] as well as the types described in section 6.1 of [RFC8008].

This document requests the registration of the two additional payload types:

Payload Type	Specification
FCI.Telemetry	RFCthis
FCI.CapacityLimits	RFCthis

Table 3

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

3.1.1. CDNI FCI Telemetry Payload Type

Purpose: The purpose of this Payload Type is to list the supported telemetry sources and the metrics made available by each source).

Interface: FCI.

Encoding: See section Section 2.1.

3.1.2. CDNI FCI Capacity Limits Payload Type

Purpose: The purpose of this Payload Type is to define Capacity Limits based on a utilization metrics corresponding to telemetry sources provided by the dCDN.

Interface: FCI.

Encoding: See section Section 2.2.

4. Security Considerations

This specification is in accordance with the CDNI Request Routing: Footprint and Capabilities Semantics. As such, it is subject to the security and privacy considerations as defined in Section 8 of [RFC8006] and in Section 7 of [RFC8008] respectively.

5. Acknowledgements

The authors would like to express their gratitude to the members of the Streaming Video Technology Alliance [SVTA] Open Caching Working Group for their guidance, contribution, and review.

6. References

6.1. 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/info/rfc2119>.

[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>.

[RFC8007] Murray, R. and B. Niven-Jenkins, "Content Delivery Network Interconnection (CDNI) Control Interface / Triggers", RFC 8007, DOI 10.17487/RFC8007, December 2016, <https://www.rfc-editor.org/info/rfc8007>.

[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>.

[RFC8804] Finkelman, O. and S. Mishra, "Content Delivery Network Interconnection (CDNI) Request Routing Extensions", RFC 8804, DOI 10.17487/RFC8804, September 2020, <https://www.rfc-editor.org/info/rfc8804>.

6.2. Informative References

[OC-CII] Ryan, A., Ed., Rosenblum, B., Goldstein, G., Roskin, R., and G. Bichot, "Open Caching Capacity Insights - Functional Specification (Placeholder before publication)", <https://www.svta.org/document/open-caching-capacity-interface/>.

[OC-RR] Finkelman, O., Ed., Hofmann, J., Klein, E., Mishra, S., Ma, K., Sahar, D., and B. Zurat, "Open Caching Request Routing - Functional Specification", Version 1.1, 4 October 2019, <https://www.svta.org/product/open-cache-request-routing-functional-specification/>.

[OCWG] "Open Caching Home Page", <https://opencaching.svta.org/>.

[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>.

[RFC7336] Peterson, L., Davie, B., and R. van Brandenburg, Ed., "Framework for Content Distribution Network Interconnection (CDNI)", RFC 7336, DOI 10.17487/RFC7336, August 2014, <https://www.rfc-editor.org/info/rfc7336>.

[SVTA] "Streaming Video Technology Alliance Home Page", <https://www.svta.org>.

Authors' Addresses

Andrew Ryan
Disney Streaming
1211 Avenue of the Americas
New York , NY 10036
United States of America

Email: andrew@andrewnryan.com

Ben Rosenblum
Vecima
4375 River Green Pkwy #100
Duluth , GA 30096
United States of America

Email: ben@rosenblum.dev

Nir B. Sopher
Qwilt
6, Ha'harash
Hod HaSharon 4524079
Israel

Email: nir@apache.org