

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 16, 2018

O. Finkelman
Qwilt
S. Mishra
Verizon
February 12, 2018

CDNI SVA Request Routing Extensions
draft-finkelman-cdni-rr-sva-extensions-00

Abstract

The Open Caching working group of the Streaming Video Alliance is focused on the delegation of video delivery requests from commercial CDNs to a caching layer at the ISP. In that aspect, Open Caching 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).

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#) [[RFC2119](#)].

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 August 16, 2018.

Copyright Notice

Copyright (c) 2018 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	2
1.1.	Terminology	2
2.	Redirect Targets Capability Object	3
2.1.	Redirect Target Address	4
3.	uCDN fallback metadata	5
3.1.	Fallback Address	6
4.	IANA Considerations	7
4.1.	CDNI Payload Types	7
4.1.1.	CDNI FCI RedirectTargets Payload Type	7
4.1.2.	CDNI MI Fallback Payload Type	7
5.	Security Considerations	8
6.	Acknowledgements	8
7.	Contributors	8
8.	References	8
8.1.	Normative References	8
8.2.	Informative References	9
	Authors' Addresses	9

[1.](#) Introduction

This document defines objects needed for Open Caching request routing. For that purpose it extends CDNI metadata [[RFC8006](#)] and CDNI Footprint and Capabilities [[RFC8008](#)]. For consistency, this document follows the CDNI notation of uCDN (the commercial CDN) and dCDN (the ISP caching layer).

The CDNI metadata interface is described in [[RFC8006](#)].

The CDNI footprint and capability interface is described in [[RFC8008](#)].

[1.1.](#) Terminology

This document reuses the terminology defined in [[RFC6707](#)], [[RFC8006](#)], [[RFC8007](#)], and [[RFC8008](#)].

Additionally, the following terms are used throughout this document and are defined as follows:

- o SVA - Streaming Video Alliance.
- o OC - SVA Open Caching.
- o RR - Request Router.
- o CP - Content Provider (video).

2. Redirect Targets Capability Object

Open Caching uses iterative request redirect as defined in [[RFC7336](#)]. In order for the uCDN to redirect to the dCDN it requires redirect target addresses. The redirect targets are defined as part of the Footprint and Capabilities interface.

Use cases

- * Footprint: The dCDN may want to have different targets per footprint. Note that a dCDN may spread across multiple geographies. This makes it easier to route client request to a nearby request router. Though this can be achieved using a single canonical name and geo DNS, that approach has limitations, for example a client may be using third party DNS resolver, making it impossible for the redirector to detect where the client is located.
- * Scaling: The dCDN may choose to scale its request routing service by deploying more request routers in new locations and advertise them via an updatable interface like the FCI.

The Redirect Target capability object is used to indicate the target addresses the uCDN should use in order to redirect a client to the uCDN. Targets are represented as endpoint objects as defined in [[RFC8006](#)].

Property: target-addresses

Description: Target addresses to which the uCDN can redirect the client, listed in order of priority.

Type: Array of target-address objects (see [Section 2.1](#))

Mandatory-to-Specify: No. The dCDN can advertise the redirect targets to the uCDN statically, or by some other means

Example of Redirect Target Capability object (which contains two target-address objects) that describes which target addresses in the dCDN the uCDN should use in order to redirect the client to the dCDN.


```
{
  "capabilities": [
    {
      "capability-type": "FCI.RedirectTargetes",
      "capability-value": {
        "target-addresses": [
          "endpoints": [
            "a.service123.dcdn.example.com",
            "b.service123.dcdn.example.com"
          ],
          "endpoints": ["c.service123.dcdn.example.com"]
        ]
      },
      "footprints": [
        <Footprint objects>
      ]
    }
  ]
}
```

2.1. Redirect Target Address

A target-address object describes the address to be used by the uCDN when redirecting a client to the dCDN.

Endpoints within a target-address object MUST be treated as equivalent/equal. A dCDN can specify an array of target-addresses, ordered by preference, within a RedirectTargets capability object. Then, for each target-address object ranked by preference, a dCDN can specify an array of endpoints that are equivalent (e.g., a pool of servers that are not behind a load balancer).

Property: endpoints

Description: Endpoint addresses to which the uCDN can redirect the client. If multiple endpoints are specified, they are all equal, i.e., the list is not ordered by preference.

Type: Array of Endpoint objects (see [section 4.3.3 of \[RFC8006\]](#)).

Mandatory-to-Specify: Yes.

Example of Target Address object (which contains two endpoint objects) that describes which endpoint addresses in the dCDN the uCDN should use in order to to redirect the client to the dCDN.


```
"endpoints": [  
  "a.service123.dcdn.example.com",  
  "b.service123.dcdn.example.com"  
]
```

3. uCDN fallback metadata

Open Caching requires that the uCDN should provide fallback servers to the dCDN to be used in cases where the dCDN cannot properly handle the request. To avoid redirect loops, the fallback servers' addresses at the uCDN MUST be different than the original address at the uCDN from which the client was redirected to the dCDN. The uCDN MUST avoid further redirection when receiving the client request at the fallback server address. The fallback server is defined as a generic metadata object (see [section 3.2 of \[RFC8006\]](#))

Use cases

- * **Failover:** A dCDN request router receives a request but has no caches to which it can route the request to. This can happen in the case of failures, or temporary network overload. In these cases, the router may choose to redirect the request back to the uCDN fallback address.
- * **Error:** A cache may receive a request that it cannot properly serve, for example, some of the metadata objects for that service were not properly acquired. In this case the cache may resolve to redirect back to uCDN.

The Fallback metadata object is used to indicate the server addresses the dCDN should use in order to redirect a client back to the uCDN. Fallbacks addresses are represented as endpoint objects as defined in [\[RFC8006\]](#).

Property: fallback-addresses

Description: Fallback Addresses to which the uCDN can redirect the client, listed in order of priority.

Type: Array of fallback-address objects (see [Section 3.1](#))

Mandatory-to-Specify: No. The dCDN can advertise the redirect addresses to the uCDN statically, or by some other means

Example of MI.Fallback Metadata object (which contains two fallback-address objects) that describes which hosts addresses in the uCDN the dCDN should use in order to redirect the client back to a fallback address at the uCDN.


```
{
  "generic-metadata-type": "MI.Fallback",
  "generic-metadata-value":
    {
      "fallback-addresses": [
        {
          "endpoints": [
            "fallback-a.service123.ucdn.example",
            "fallback-b.service123.ucdn.example"
          ],
          "protocol": "http/1.1"
        },
        {
          "endpoints": ["fallback-c.service123.example"],
          "protocol": "http/1.1"
        }
      ]
    }
}
```

3.1. Fallback Address

A fallback-address object describes the address to be used by the dCDN when redirecting a client back to the dCDN due to failure, error, or other conditions in the dCDN.

Endpoints within a fallback-address object MUST be treated as equivalent/equal. A uCDN can specify an array of fallback-addresses, ordered by preference, within a Fallback metadata object. Then, for each fallback-address object ranked by preference, a uCDN can specify an array of endpoints that are equivalent (e.g., a pool of servers that are not behind a load balancer).

Property: endpoints

Description: Endpoint addresses to which the dCDN can redirect the client. If multiple endpoints are specified, they are all equal, i.e., the list is not ordered by preference..

Type: Array of Endpoint objects (see [section 4.3.3 of \[RFC8006\]](#))

Mandatory-to-Specify: Yes.

Property: protocol

Description: Network protocol to use when redirecting to this fallback server.

Type: Protocol (see [section 4.3.2 of \[RFC8006\]](#))

Mandatory-to-Specify: Yes.

Example of Fallback Address object (which contains two endpoint objects) that describes which endpoint addresses in the uCDN the dCDN should use in order to to redirect the client to the uCDN.

```
{
  "endpoints": [
    "fallback-a.service123.ucdn.example",
    "fallback-b.service123.ucdn.example"
  ],
  "protocol": "http/1.1"
}
```

4. IANA Considerations

4.1. CDNI Payload Types

This document requests the registration of the following CDNI Payload Types under the IANA CDNI Payload Type registry [[RFC7736](#)]:

+-----+	+-----+
Payload Type	Specification
+-----+	+-----+
FCI.RedirectTargetes	RFCthis
MI.Fallback	RFCthis
+-----+	+-----+

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

4.1.1. CDNI FCI RedirectTargets Payload Type

Purpose: The purpose of this payload type is to distinguish RedirectTargets FCI objects

Interface: FCI

Encoding: see [Section 2](#)

4.1.2. CDNI MI Fallback Payload Type

Purpose: The purpose of this payload type is to distinguish Fallback MI objects (and any associated capability advertisement)

Interface: MI/FCI

Encoding: see [Section 3](#)

5. Security Considerations

This specification is in accordance with the CDNI Metadata Interface and the CDNI Request Routing: Footprint and Capabilities Semantics. As such, it is subject to the security considerations as defined in [\[RFC8006\]](#) and [\[RFC8008\]](#) respectively.

6. Acknowledgements

TBD.

7. Contributors

TBD.

8. References

8.1. 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>>.
- [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>>.
- [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>>.

8.2. Informative References

[RFC7736] Ma, K., "Content Delivery Network Interconnection (CDNI) Media Type Registration", [RFC 7736](#), DOI 10.17487/RFC7736, December 2015, <<https://www.rfc-editor.org/info/rfc7736>>.

Authors' Addresses

Ori Finkelman
Qwilt
6, Ha'harash
Hod HaSharon 4524079
Israel

Phone: +972-72-2221647
Email: orif@qwilt.com

Sanjay Mishra
Verizon
13100 Columbia Pike
Silver Spring, MD 20904
USA

Email: sanjay.mishra@verizon.com

