ï»¿

Internet Engineering Task Force (IETF)
Request for Comments: 8804
Category: Standards Track
ISSN: 2070-1721

O. Finkelman
Qwilt
S. Mishra
Verizon

September 2020

Content Delivery Network Interconnection (CDNI) Request Routing Extensions

Abstract

Open Caching architecture is a use case of Content Delivery Network 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 defines extensions to the CDNI Metadata Interface (MI) and the Footprint & Capabilities Advertisement interface (FCI). These extensions are derived from requirements raised by Open Caching but are also applicable to CDNI use cases in general.

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/rfc8804.

Table of Contents

1. Introduction

The Streaming Video Alliance [SVA] 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 Alliance [SVA] 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). The Open Caching Request Routing Functional Specification [OC-RR] defines the Request Routing process and the interfaces that are required for its provisioning. This document defines the CDNI metadata object [RFC8006] and the CDNI Footprint and Capabilities object [RFC8008] that are required for Open Caching Request Routing:

* Redirect Target Capability (for dCDN advertising redirect target address)

* Fallback Target Metadata (for uCDN configuring fallback target address)

This document also registers CDNI Payload Types [RFC7736] for these defined objects.

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.

1.1. Terminology

The following terms are used throughout this document:

FQDN Fully Qualified Domain Name

CDN Content Delivery Network

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

FCI Footprint & Capabilities Advertisement interface (see [RFC8008])

MI Metadata Interface (see [RFC8006])

uCDN Upstream CDN (see [RFC7336])

dCDN Downstream CDN (see [RFC7336])

RT Redirection Target. Endpoint for redirection from uCDN to dCDN.

RR Request Router. An element responsible for routing user requests, typically using HTTP redirect or DNS CNAME, depending on the use case.

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.

2. Redirect Target Capability

Iterative CDNI Request Redirection is defined in Section 1.1 of [RFC7336] and elaborated by examples in Sections 3.2 and 3.4 of [RFC7336]. A Redirection Target (RT) is defined in Section 2 of [RFC7975] for Recursive Request Redirection as:

> The endpoint to which the User Agent is redirected. In CDNI, an RT may point to a number of different components, some examples include a surrogate in the same CDN as the request router, a request router in a dCDN, or a surrogate in a dCDN.

In this document, we adopt the same definition of the RT for the Iterative Request Redirect use case. This use case requires the provisioning of the RT address to be used by the uCDN in order to redirect to the dCDN. RT addresses can vary between different footprints (for example, between different regions), and they may also change over time (for example, as a result of network problems). Given this variable and dynamic nature of the redirect target address, it may not be suitable to advertise it during bootstrap. A more dynamic and footprint-oriented interface is required. Section 4.3 of [RFC7336] suggests that it could be one of the roles of the FCI [RFC8008]. Following this suggestion, we have therefore chosen to use the CDNI Footprint & Capabilities Advertisement interface for redirect target address advertisement.

Use cases:

* Footprint: The dCDN may want to have a different target per footprint. Note that a dCDN may spread across multiple geographies. This makes it easier to route client requests to a nearby request router. Though this can be achieved using a single canonical name and "Geo DNS", such that in different geographies the same hostname is resolved to different IP address, that approach has limitations; for example, a client may be using a third-party DNS resolver, making it impossible for the redirector to detect where the client is located, or Geo DNS granularity may be too rough for the requirement of the application.

* 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 address the uCDN should use in order to redirect a client to the dCDN. A target may be attached to a specific uCDN host, attached to a list of uCDN hosts, or used globally for all the hosts of the uCDN.

When a dCDN is attaching the redirect target to a specific uCDN host or a list of uCDN hosts, the dCDN MUST advertise the hosts within the Redirect Target capability object as "redirecting-hosts". In this case, the uCDN can redirect to that dCDN address, only if the User Agent request was to one of these uCDN hosts.

If the Redirect Target capability object does not contain a target or the target is empty, the uCDN MUST interpret it as "no target available for these uCDN hosts for the specified footprint". In case such a target was already advertised in a previous FCI object, the uCDN MUST interpret it as an update that deletes the previous redirect target.

2.1. DNS Redirect Target

A redirect target for DNS redirection is an FQDN used as an alias in

a CNAME record response (see [RFC1034]) of the uCDN DNS router. Note that DNS routers make routing decisions based on either the DNS resolver's IP address or the client IP subnet when EDNS0 client-subnet (ECS) is used (see [RFC7871]). The dCDN may choose to advertise redirect targets and footprints to cover both cases, such that the uCDN resolution would route the DNS query to different dCDN CNAMEs according to client subnet or dCDN resolver IP address. This method further allows the dCDN DNS to optimize the resolution by localizing the target CNAMEs. A uCDN implementation SHOULD prefer routing based on client IP subnet when the ECS option is present. A dCDN implementation using the ECS option MUST be aware of the privacy drawbacks listed in Section 2 of [RFC7871] and SHOULD follow the guidelines provided in Section 11.1 of [RFC7871].

2.2. HTTP Redirect Target

A redirect target for HTTP redirection is the URI to be used as the value for the Location header of an HTTP redirect 3xx response, typically a 302 (Found) (see Section 7.1.2 of [RFC7231] and Section 6.4 of [RFC7231]).

2.3. Properties of Redirect Target Capability Object

The Redirect Target capability object consists of the following properties:

Property: redirecting-hosts

Description: One or more uCDN hosts to which this redirect target is attached. A redirecting host SHOULD be a host that was published in a HostMatch object by the uCDN as defined in Section 4.1.2 of [RFC8006].

Type: A list of Endpoint objects (see Section 4.3.3 of [RFC8006])

Mandatory-to-Specify: No. If absent or empty, the redirect target applies to all hosts of the redirecting uCDN.

Property: dns-target

Description:
Target CNAME record for DNS redirection.

Type:
DnsTarget object (see Section 2.4)

Mandatory-to-Specify:
No. If the dns-target is absent or empty, the uCDN MUST interpret it as "no dns-target available".

Property: http-target

Description:
Target URI for an HTTP redirect.

Type:
HttpTarget object (see Section 2.5)

Mandatory-to-Specify:
No. If the http-target is absent or empty, the uCDN MUST interpret it as "no http-target available".

The following is an example of a Redirect Target capability object serialization that advertises a dCDN target address that is attached to a specific list of uCDN "redirecting-hosts". A uCDN host that is included in that list can redirect to the advertised dCDN redirect target. The capabilities object is serialized as a JSON object as defined in Section 5.1 of [RFC8008].

```
{
  "capabilities": [
```

```
      {
        "capability-type": "FCI.RedirectTarget",
        "capability-value": {
            "redirecting-hosts": [
               "a.service123.ucdn.example.com",
               "b.service123.ucdn.example.com"
            ],
            "dns-target": {
               "host": "service123.ucdn.dcdn.example.com"
            },
            "http-target": {
                "host": "us-east1.dcdn.example.com",
                "path-prefix": "/cache/1/",
                "include-redirecting-host": true
            }
        },
        "footprints": [
            <Footprint objects>
        ]
      }
    ]
  }
```

2.4. DnsTarget Object

The DnsTarget object gives the target address for the DNS response to delegate from the uCDN to the dCDN.

Property: host

Description: The host property is a hostname or an IP address, without a port number.

Type: Endpoint object as defined in Section 4.3.3 of [RFC8006], with the limitation that it SHOULD NOT include a port number and, in case a port number is present, the uCDN MUST ignore it.

Mandatory-to-Specify: Yes.

2.4.1. DnsTarget Example

The following is an example of the DnsTarget object:

```
{
   "host": "service123.ucdn.dcdn.example.com"
}
```

The following is an example of a DNS query for uCDN address "a.service123.ucdn.example.com" and the corresponding CNAME redirection response:

```
Query:
a.service123.ucdn.example.com:
type A, class IN

Response:
NAME: a.service123.ucdn.example.com, TYPE: CNAME, CLASS: IN,
TTL: 120, RDATA: service123.ucdn.dcdn.example.com
```

2.5. HttpTarget Object

The HttpTarget object gives the necessary information to construct the target Location URI for HTTP redirection.

Property: host

Description: Hostname or IP address and an optional port, i.e., the host and port of the authority component of the URI as described in Section 3.2 of [RFC3986].

Type: Endpoint object as defined in Section 4.3.3 of [RFC8006].

Mandatory-to-Specify: Yes.

Property: scheme

Description: A URI scheme to be used in the redirect response location construction. When present, the uCDN MUST use the provided scheme in for HTTP redirection to the dCDN.

Type: A URI scheme as defined in Section 3.1 of [RFC3986], represented as a JSON string. The scheme MUST be either "http" or "https".

Mandatory-to-Specify: No. If this property is absent or empty, the uCDN request router MUST use the same scheme as was used in the original request before redirection.

Property: path-prefix

Description: A path prefix for the HTTP redirect Location header. The original path is appended after this prefix.

Type: A prefix of a path-absolute as defined in Section 3.3 of [RFC3986]. The prefix MUST end with a trailing slash to indicate the end of the last path segment in the prefix.

Mandatory-to-Specify: No. If this property is absent or empty, the uCDN MUST NOT prepend a path-prefix to the original content path, i.e., the original path MUST appear in the Location URI right after the authority component.

Property: include-redirecting-host

Description: A flag indicating whether or not to include the redirecting host as the first path segment after the path-prefix. If set to true and a "path-prefix" is used, the uCDN redirecting host MUST be added as a separate path segment after the path-prefix and before the original URL path. If set to true and there is no path-prefix, the uCDN redirecting host MUST be prepended as the first path segment in the redirect URL.

Type: Boolean.

Mandatory-to-Specify: No. Default value is False.

2.5.1. HttpTarget Example

The following is an example of the HttpTarget object with a "scheme", a "path-prefix", and "include-redirecting-host" properties:

```
{
   "host": "us-east1.dcdn.example.com",
   "scheme": "https",
   "path-prefix": "/cache/1/",
   "include-redirecting-host": true
}
```

The following is an example of an HTTP request for content at uCDN host "a.service123.ucdn.example.com" and the corresponding HTTP response with a Location header, used for redirecting the client to the dCDN, constructed according to the HttpTarget object from the above example:

```
Request:
GET /vod/1/movie.mp4 HTTP/1.1
Host: a.service123.ucdn.example.com

Response:
HTTP/1.1 302 Found
Location: https://us-east1.dcdn.example.com/cache/1/
```

a.service123.ucdn.example.com/vod/1/movie.mp4

2.6. Usage Example

Before requests can be routed from the uCDN to the dCDN, the CDNs must exchange service configurations between them. Using the MI, the uCDN advertises out-of-band its hosts to the dCDN; each host is designated by a hostname and has its own specific metadata (see Section 4.1.2 of [RFC8006]). Using the FCI, the dCDN advertises (also out-of-band) the redirect target address defined in Section 2.3 for the relevant uCDN hosts. The following is a generalized example of the message flow between a uCDN and a dCDN. For simplicity, we focus on the sequence of messages between the uCDN and dCDN and not on how they are passed.

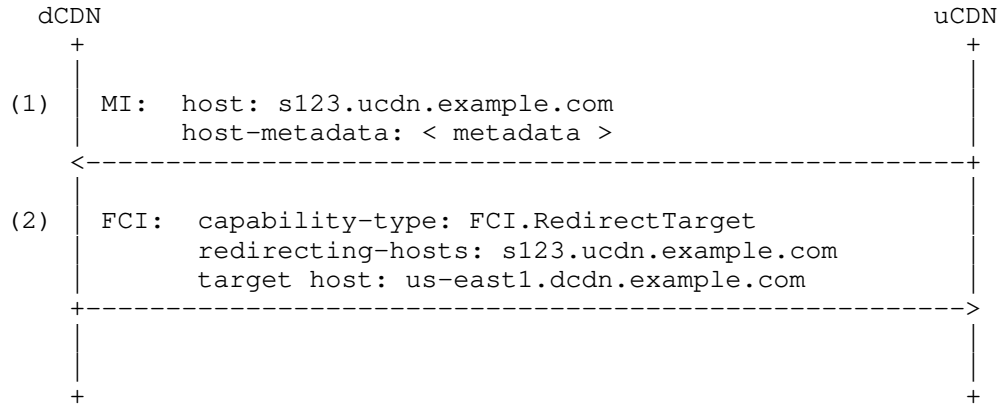

Figure 1: Redirect Target Address Advertisement

Explanation:

(1) The uCDN advertises a host (s123.ucdn.example.com) with the host metadata.

(2) The dCDN advertises its FCI objects to the uCDN, including a Redirect Target capability object that contains the redirect target address (us-east1.dcdn.example.com) specified for that uCDN host.

Once the redirect target has been set, the uCDN can start redirecting user requests to the dCDN. The following is a generic sequence of redirection using the host and redirect target that were advertised in Figure 1.

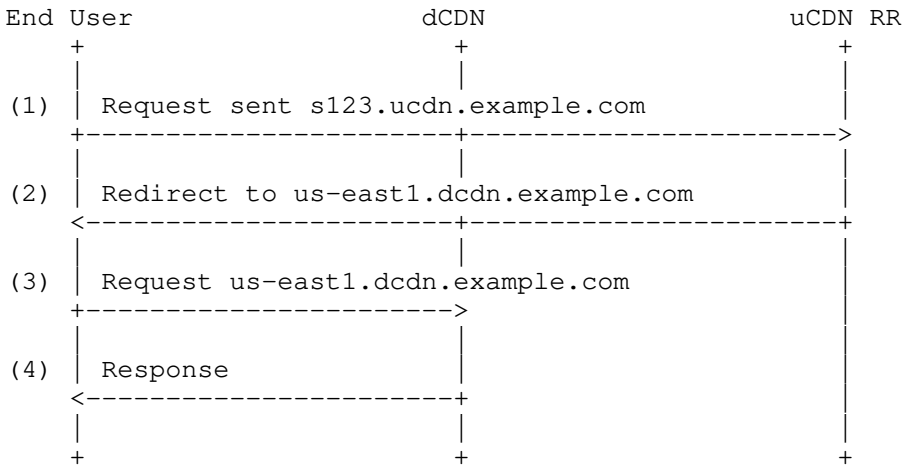

Figure 2: Generic Request Redirection Sequence

Explanation:

(1) The End User sends a request (DNS or HTTP) to the uCDN Request Router (RR).

(2) Using the previously advertised Redirect Target, the uCDN redirects the request to the dCDN.

(3) The End User sends a request to the dCDN.

(4) The dCDN either sends a response or reroutes it, for example, to a dCDN surrogate.

3. Fallback Target Server Address

Open Caching requires that the uCDN provides a fallback target server to the dCDN to be used in cases where the dCDN cannot properly handle the request. To avoid redirect loops, the fallback target server's address at the uCDN MUST be different from the original uCDN address from which the client was redirected to the dCDN. The uCDN MUST avoid further redirection when receiving the client request at the fallback target. The Fallback Target 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. This can happen in the case of failures or temporary network overload.

* No coverage: A dCDN request router receives a request from a client located in an area inside the footprint but not covered by the dCDN caches or outside the dCDN footprint coverage. In such 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's "default action" may be to "redirect back to uCDN".

The Fallback Target metadata object is used to indicate the target address the dCDN should redirect a client to when falling back to the uCDN. The fallback target address is represented as an Endpoint object as defined in Section 4.3.3 of [RFC8006].

In DNS redirection, a CNAME record is used as the fallback target address.

In HTTP redirection, a hostname is used as the fallback target address.

When using HTTP redirect to route a client request back to the uCDN, it is the dCDN's responsibility to use the original URL path as the client would have used for the original uCDN request, stripping, if needed, the dCDN path-prefix and/or the uCDN hostname from the redirect URL that may have been used to request the content from the dCDN.

3.1. Properties of Fallback Target Generic Metadata Object

The MI.FallbackTarget generic metadata object consists of the following two properties:

Property: host

Description: Target address to which the dCDN can redirect the client.

Type: Endpoint object as defined in Section 4.3.3 of [RFC8006], with the limitation that in case of DNS delegation, it SHOULD NOT include a port number, and in case a port number is present, the dCDN MUST ignore it.

Mandatory-to-Specify: Yes.

Property: scheme

Description: A URI scheme to be used in the redirect response location construction. When present, the dCDN MUST use this scheme in case of HTTP redirection to the uCDN fallback address.

Type: A URI scheme as defined in Section 3.1 of [RFC3986], represented as a JSON string. The scheme MUST be either "http" or "https".

Mandatory-to-Specify: No. In case of HTTP redirection to fallback, if this property is absent or empty, the dCDN redirecting entity MUST use the same scheme as in the request received by the dCDN.

The following is an example of an MI.FallbackTarget generic metadata object that designates the host address the dCDN should use as fallback address to redirect back to the uCDN:

```
{
    "generic-metadata-type": "MI.FallbackTarget",
    "generic-metadata-value":
    {
        "host": "fallback-a.service123.ucdn.example",
        "scheme": "https"
    }
}
```

3.2. Usage Example

The uCDN advertises out-of-band the fallback target address to the dCDN, so that the dCDN may redirect a request back to the uCDN in case the dCDN cannot serve it. Using the MI, the uCDN advertises its hosts to the dCDN, along with their specific host metadata (see Section 4.1.2 of [RFC8006]). The Fallback Target generic metadata object is encapsulated within the "host-metadata" property of each host. The following is an example of a message flow between a uCDN and a dCDN. For simplicity, we focus on the sequence of messages between the uCDN and dCDN, not on how they are passed.

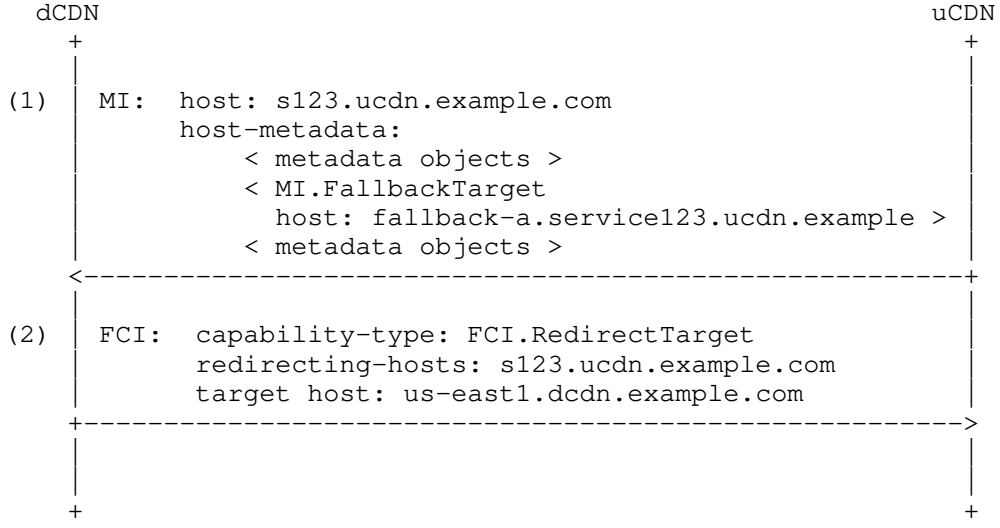

Figure 3: Advertisement of Host Metadata with Fallback Target

Explanation:

(1) The uCDN advertises a host (s123.ucdn.example.com) with the host metadata. The host-metadata property contains an MI.FallbackTarget generic metadata object.

(2) The dCDN advertises its FCI objects to the uCDN, including a Redirect Target capability object that contains the redirect target address (us-east1.dcdn.example.com) specified for that

uCDN host.

The following is a generic sequence of redirection using the configurations that were advertised in Figure 3. In this case, the dCDN redirects back to the uCDN fallback target address.

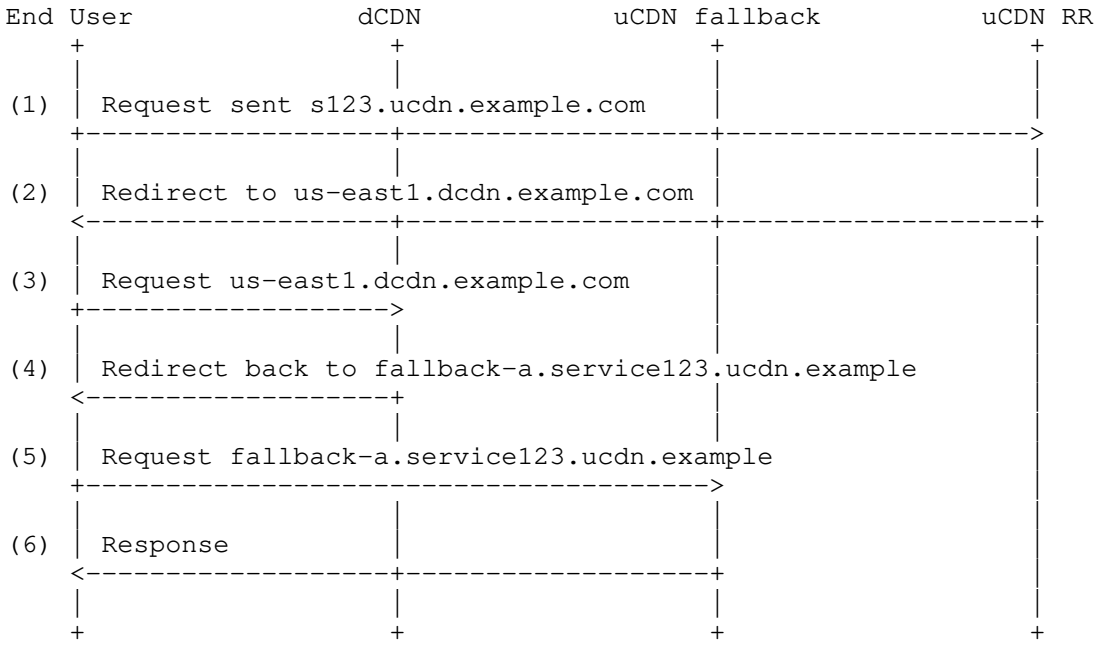

Figure 4: Redirection to Fallback Target

Explanation:

(1) The End User sends a request (DNS or HTTP) to the uCDN Request Router (RR).

(2) Using the previously advertised Redirect Target, the uCDN redirects the request to the dCDN.

(3) The End User sends a request to the dCDN.

(4) The dCDN cannot handle the request and therefore redirects it back to the uCDN fallback target address.

(5) The End User sends the request to the uCDN fallback target address.

(6) The uCDN either sends a response or reroutes it, for example, to a uCDN surrogate.

3.3. uCDN Addressing Considerations

When advertising fallback addresses to the dCDN, the uCDN SHOULD consider the failure use cases that may lead the dCDN to route requests to uCDN fallback. In extreme dCDN network failures or under denial-of-service (DoS) attacks, requests coming from a large segment or multiple segments of the dCDN may be routed back to the uCDN. The uCDN SHOULD therefore design its fallback addressing scheme and its available resources accordingly. A favorable approach would be for the uCDN to use a different fallback target address for each uCDN host, enabling it to load balance the requests using the same methods as it would for its original hosts. See Sections 4.1.2 and 4.1.3 of [RFC8006] for a detailed description of how to use GenericMetadata objects within the HostMatch object advertised in the HostIndex of the uCDN.

4. IANA Considerations

4.1. CDNI Payload Types

IANA has registered the following CDNI Payload Types in the "CDNI

Payload Types" registry defined in [RFC7736]:

Payload Type	Specification
FCI.RedirectTarget	RFC 8804
MI.FallbackTarget	RFC 8804

Table 1

4.1.1. CDNI FCI RedirectTarget Payload Type

Purpose: The purpose of this payload type is to distinguish FCI advertisement objects for redirect target.

Interface: FCI

Encoding: See Section 2.3.

4.1.2. CDNI MI FallbackTarget Payload Type

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

Interface: MI/FCI

Encoding: See Section 3.1.

5. Security Considerations

This specification defines extensions to the CDNI Metadata Interface (MI) and the Footprint & Capabilities Advertisement interface (FCI). As such, it is subject to the security and privacy considerations defined in Section 8 of [RFC8006] and in Section 7 of [RFC8008], respectively.

5.1. Confidentiality and Privacy

The Redirect Target capability object potentially reveals information about the internal structure of the dCDN network. A third party could intercept the FCI transactions and use the information to attack the dCDN. The same is also true for the Fallback Target generic metadata object, as it may reveal information about the internal structure of the uCDN, exposing it to external exploits. Implementations of the FCI and MI MUST therefore use strong authentication and encryption and strictly follow the directions for securing the interface as defined for the Metadata Interface in Section 8.3 of [RFC8006].

6. References

6.1. Normative References

[RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <https://www.rfc-editor.org/info/rfc1034>.

[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>.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <https://www.rfc-editor.org/info/rfc3986>.

[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>.

[RFC7231] Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content", RFC 7231, DOI 10.17487/RFC7231, June 2014, <https://www.rfc-editor.org/info/rfc7231>.

[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>.

[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>.

[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>.

6.2. Informative References

[OC-RR] Finkelman, O., Ed., Hofmann, J., Klein, E., Mishra, S., Ma, K., Sahar, D., and B. Zurat, "Open Cache Request Routing Functional Specification", Version 1.1, November 2016, <https://www.streamingvideoalliance.org/books/open-cache-request-routing-functional-specification/>.

[OCWG] Streaming Video Alliance, "Open Caching", <https://www.streamingvideoalliance.org/technical-groups/open-caching/>.

[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>.

[RFC7871] Contavalli, C., van der Gaast, W., Lawrence, D., and W. Kumari, "Client Subnet in DNS Queries", RFC 7871, DOI 10.17487/RFC7871, May 2016, <https://www.rfc-editor.org/info/rfc7871>.

[SVA] "Streaming Video Alliance", <https://www.streamingvideoalliance.org>.

Acknowledgements

The authors thank Nir B. Sopher for reality checks against production use cases; his contribution is significant to this document. The authors also thank Ben Niven-Jenkins for his review and feedback and Kevin J. Ma for his guidance throughout the development of this document, including his regular reviews.

Authors' Addresses

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

Email: ori.finkelman.ietf@gmail.com

Sanjay Mishra
Verizon
13100 Columbia Pike
Silver Spring, MD 20904
United States of America

Email: sanjay.mishra@verizon.com