

Workgroup: HTTP  
Internet-Draft:  
draft-ietf-httpbis-targeted-cache-control-00  
Published: 26 July 2021  
Intended Status: Standards Track  
Expires: 27 January 2022  
Authors: S. Ludin    M. Nottingham    Y. Wu  
         Akamai       Fastly           Cloudflare  
**Targeted HTTP Cache Control**

## Abstract

This specification defines a convention for HTTP response header fields that allow directives controlling caching to be targeted at specific caches or classes of caches. It also defines one such header field, targeted at Content Delivery Network (CDN) caches.

## Note to Readers

*RFC EDITOR: please remove this section before publication*

The issues list for this draft can be found at <https://github.com/httpwg/http-extensions/labels/targeted-cc>.

The most recent (often, unpublished) draft is at <https://httpwg.org/http-extensions/draft-ietf-httpbis-targeted-cache-control.html>.

See also the draft's current status in the IETF datatracker, at <https://datatracker.ietf.org/doc/draft-ietf-httpbis-targeted-cache-control/>.

## Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 27 January 2022.

## Copyright Notice

Copyright (c) 2021 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](#)
  - [1.1. Notational Conventions](#)
- [2. Targeted Cache-Control Header Fields](#)
  - [2.1. Cache Behavior](#)
  - [2.2. Parsing Targeted Fields](#)
  - [2.3. Defining Targeted Fields](#)
- [3. The CDN-Cache-Control Targeted Field](#)
  - [3.1. Examples](#)
- [4. IANA Considerations](#)
- [5. Security Considerations](#)
- [6. Normative References](#)
- [Authors' Addresses](#)

### 1. Introduction

Modern deployments of HTTP often use multiple layers of caching with varying properties. For example, a Web site might use a cache on the origin server itself; it might deploy a caching layer in the same network as the origin server, it might use one or more Content Delivery Networks (CDNs) that are distributed throughout the Internet, and it might utilise browser caching as well.

Because it is often desirable to control these different classes of caches separately, some means of targeting directives at them is necessary.

The HTTP Cache-Control response header field is widely used to direct caching behavior. However, it is relatively undifferentiated; while some directives (e.g., s-maxage) are targeted at a specific class of caches (for s-maxage, shared caches), that is not consistently available across all existing cache directives (e.g., stale-while-revalidate). This is problematic, especially as the

number of caching extensions grows, along with the number of potential targets.

Some caches have defined ad hoc control mechanisms to overcome this issue, but interoperability is low. [Section 2](#) defines a standard framework for targeted cache control using HTTP response headers, and [Section 3](#) defines one such header: the CDN-Cache-Control response header field.

### 1.1. Notational Conventions

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. Targeted Cache-Control Header Fields

A Targeted Cache-Control Header Field (hereafter, "targeted field") is a HTTP response header field that has the same syntax and semantics as the Cache-Control response header field [[I-D.ietf-httpbis-cache](#)], [Section 5.2](#). However, it has a distinct field name that indicates the target for its directives.

For example:

```
CDN-Cache-Control: max-age=60
```

is a targeted field that applies to Content Delivery Networks (CDNs), as defined in [Section 3](#).

### 2.1. Cache Behavior

A cache that implement this specification has a *target list* - an ordered list of the targeted field names that it uses for caching policy, with the order reflecting priority from most applicable to least. The target list might be fixed, user-configurable, or generated per request, depending upon the implementation.

For example, a CDN cache might support both CDN-Cache-Control and a header specific to that CDN, ExampleCDN-Cache-Control, with the latter overriding the former. Its target list would be:

```
[ExampleCDN-Cache-Control, CDN-Cache-Control]
```

When a cache that implements this specification receives a response with one or more of the header field names on its target list,

the cache MUST select the first (in target list order) field with a valid, non-empty value and use that to determine the caching policy for the response, and MUST ignore the Cache-Control and Expires header fields in that response, unless no valid, non-empty value is available from the listed header fields.

Note that this is on a response-by-response basis; if no applicable targeted field is present, valid and non-empty, a cache falls back to other cache control mechanisms as required by HTTP [[I-D.ietf-httpbis-cache](#)].

Targeted fields that are not on a cache's target list MUST NOT change that cache's behaviour, and MUST be passed through.

Caches that use a targeted field MUST implement the semantics of the following cache directives:

- \*max-age

- \*must-revalidate

- \*no-store

- \*no-cache

- \*private

Furthermore, they SHOULD implement other cache directives (including extension cache directives) that they support in the Cache-Control response header field.

The semantics and precedence of cache directives in a targeted field are the same as those in Cache-Control. In particular, no-store and no-cache make max-age inoperative.

## 2.2. Parsing Targeted Fields

Targeted fields MAY be parsed as a Dictionary Structured Field [[RFC8941](#)], and implementations are encouraged to use a parser for that format in the interests of robustness, interoperability and security.

When an implementation parses a targeted field as a Structured Field, each cache directive will be assigned a value. For example, max-age has an integer value; no-store's value is boolean true, and no-cache's value can either be boolean true or a list of field names. Implementations SHOULD NOT accept other values (e.g. coerce a max-age with a decimal value into an integer). Likewise, implementations SHOULD ignore parameters on directives, unless otherwise specified.

However, implementers MAY reuse a Cache-Control parser for simplicity. If they do so, they SHOULD observe the following points, to aid in a smooth transition to a full Structured Field parser and prevent interoperability issues:

- \*If a directive is repeated in the field value (e.g., "max-age=30, max-age=60"), the last value 'wins' (60, in this case)

- \*Members of the directives can have parameters (e.g., "max-age=30;a=b;c=d"), which should be ignored unless specified.

If a targeted field in a given response is empty, or a parsing error is encountered (when being parsed as a Structured Field), that field SHOULD be ignored by the cache (i.e., it should behave as if the field were not present, likely falling back to other cache control mechanisms present).

### 2.3. Defining Targeted Fields

A targeted field for a particular class of cache can be defined by requesting registration in the Hypertext Transfer Protocol (HTTP) Field Name Registry <https://www.iana.org/assignments/http-fields/>, listing this specification as the specification document. The Comments field of the registration SHOULD clearly define the class of caches that the targeted field applies to.

By convention, targeted fields SHOULD have the suffix "-Cache-Control": e.g., "ExampleCDN-Cache-Control". However, this suffix MUST NOT be used on its own to identify targeted fields; it is only a convention.

## 3. The CDN-Cache-Control Targeted Field

The CDN-Cache-Control response header field is a targeted field [Section 2](#) that allows origin servers to control the behaviour of CDN caches interposed between them and clients, separately from other caches that might handle the response.

It applies to caches that are part of a distributed network that operate on behalf of an origin server (commonly called a Content Delivery Network or CDN).

CDN caches that use CDN-Cache-Control MAY forward this header so that downstream CDN caches can use it as well. However, doing so exposes its value to all downstream clients, which might be undesirable. As a result, CDN caches that process this header field MAY remove it (for example, when configured to do so because it is known not to be used downstream).

### 3.1. Examples

For example, the following header fields would instruct a CDN cache to consider the response fresh for 600 seconds, other shared caches for 120 seconds and any remaining caches for 60 seconds:

```
Cache-Control: max-age=60, s-maxage=120
CDN-Cache-Control: max-age=600
```

These header fields would instruct a CDN cache to consider the response fresh for 600 seconds, while all other caches would be prevented from storing it:

```
Cache-Control: no-store
CDN-Cache-Control: max-age=600
```

Because CDN-Cache-Control is not present, this header field would prevent all caches from storing the response:

```
Cache-Control: no-store
```

Whereas these would prevent all caches except for CDN caches from storing the response:

```
Cache-Control: no-store
CDN-Cache-Control: none
```

(note that 'none' is not a registered cache directive; it is here to avoid sending a header field with an empty value, because such a header might not be preserved in all cases)

### 4. IANA Considerations

Please register the following entry in the Hypertext Transfer Protocol (HTTP) Field Name Registry defined by [[I-D.ietf-httpbis-semantics](#)]:

```
*Field Name: CDN-Cache-Control

*Status: permanent

*Specification Document: [this document]
```

\*Comments: Cache-Control directives targeted at Content Delivery Networks

## 5. Security Considerations

The security considerations of HTTP caching [[I-D.ietf-httpbis-cache](#)] apply.

The ability to carry multiple caching policies on a response can result in confusion about how a response will be cached in different systems, if not used carefully. This might result in unintentional reuse of responses with sensitive information.

## 6. Normative References

[[I-D.ietf-httpbis-cache](#)] Fielding, R. T., Nottingham, M., and J. Reschke, "HTTP Caching", Work in Progress, Internet-Draft, draft-ietf-httpbis-cache-17, 25 July 2021, <<https://datatracker.ietf.org/doc/html/draft-ietf-httpbis-cache-17>>.

[[I-D.ietf-httpbis-semantic](#)s] Fielding, R. T., Nottingham, M., and J. Reschke, "HTTP Semantics", Work in Progress, Internet-Draft, draft-ietf-httpbis-semantic-17, 25 July 2021, <<https://datatracker.ietf.org/doc/html/draft-ietf-httpbis-semantic-17>>.

[[RFC2119](#)] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/rfc/rfc2119>>.

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

[[RFC8941](#)] Nottingham, M. and P-H. Kamp, "Structured Field Values for HTTP", RFC 8941, DOI 10.17487/RFC8941, February 2021, <<https://www.rfc-editor.org/rfc/rfc8941>>.

## Authors' Addresses

Stephen Ludin  
Akamai

Email: [sludin@ludin.org](mailto:sludin@ludin.org)

Mark Nottingham  
Fastly

Prahran  
VIC  
Australia

Email: [mnot@mnot.net](mailto:mnot@mnot.net)  
URI: <https://www.mnot.net/>

Yuchen Wu  
Cloudflare

Email: [me@yuchenwu.net](mailto:me@yuchenwu.net)