Network Working Group
Internet-Draft
Intended status: Informational
Expires: November 10, 2008

M. Nottingham
Yahoo! Inc.

May 9, 2008

The stale-if-error HTTP Cache-Control Extension
draft-nottingham-http-stale-if-error-01

Status of this Memo

Abstract

The stale-if-error HTTP Cache-Control extension improves availability of some kinds of cached content by allowing servers and clients to instruct caches to use stale responses when certain error conditions are encountered.

Table of Contents

1. Introduction

HTTP [RFC2616] requires that caches "respond to a request with the most up-to-date response held... that is appropriate to the request," although "in carefully controlled circumstances" a stale response is allowed to be returned.

Those circumstances are not well-defined. Often, it is useful to return a stale response when an error -- e.g., a 500 Internal Server Error, a network segment, or DNS failure -- is encountered, but caches are understandably reluctant to act without explicit instructions about the appropriate behaviour.

The stale-if-error HTTP Cache-Control extension addresses this by allowing origin servers as well as clients to instruct caches to use a stale response under certain conditions, rather than returning a "hard" error, thus improving availability.

2. Notational Conventions

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].

This specification uses the augmented Backus-Naur Form of RFC2616 [RFC2616], and includes the delta-seconds rule from that specification.

3. The stale-if-error Cache-Control Extension

The stale-if-error Cache-Control extension indicates that when an error is encountered, a cached stale response MAY be used to satisfy the request, regardless of other freshness information.

```
stale-if-error = "stale-if-error" "=" delta-seconds
```

When used as a request Cache-Control extension, its scope of application is the request it appears in; when used as a response Cache-Control extension, its scope is any request applicable to the cached response it occurs in.

Its value indicates the upper limit to staleness; when the cached response is more stale than the indicated amount, the cached response MUST NOT be used to satisfy the request, absent other information.

In this context, an error is any situation which would result in a

500, 502, 503 or 504 HTTP response status code being returned.

Note that this directive only affects the freshness of a response in an implementation that recognises it; stale cached responses that are used SHOULD still be visibly stale when sent.

4. Response stale-if-error Example

A response containing:

```
HTTP/1.1 200 OK
Cache-Control: max-age=600, stale-if-error=1200
Content-Type: text/plain

success
```

indicates that it is fresh for 600 seconds, and that it may be used if an error is encountered after becoming stale for an additional 1200 seconds.

Thus, if the cache attempts to validate 900 seconds afterwards and encounters:

```
HTTP/1.1 500 Internal Server Error
Content-Type: text/plain

failure
```

the successful response can be returned instead:

```
HTTP/1.1 200 OK
Cache-Control: max-age=600, stale-if-error=1200
Age: 900
Content-Type: text/plain

succcess
```

After the age is greater than 1800 seconds (i.e., it has been stale for 1200 seconds), the cache must write the error message through.

```
HTTP/1.1 500 Internal Server Error
Content-Type: text/plain

failure
```

5. Security Considerations

This document provides origin servers and clients a mechanism for dictating that stale content should be served from caches under certain circumstances, and does not pose additional security considerations over those of RFC2616, which also allows stale content to be served.

6. IANA Considerations

This document has no actions for IANA.

7. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999.

Appendix A. Acknowledgements

Thanks to John Nienart, Henrik Nordstrom, Evan Torrie, and Chris Westin for their suggestions. The author takes all responsibility for errors and omissions.

Author's Address

Mark Nottingham
Yahoo! Inc.

Email: mnot@yahoo-inc.com
URI: http://www.mnot.net/

Full Copyright Statement

Intellectual Property