

**The "safe" HTTP Preference
draft-nottingham-safe-hint-06**

Abstract

This specification defines a "safe" preference for HTTP requests, expressing a desire to avoid "objectionable" content.

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 <http://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 17, 2015.

Copyright Notice

Copyright (c) 2015 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 (<http://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
2.	The "safe" Preference	3
3.	Implementation Status	4
4.	Security Considerations	4
5.	IANA Considerations	5
6.	References	5
Appendix A.	Acknowledgements	6
Appendix B.	Setting "safe" from Web Browsers	6
Appendix C.	Supporting "safe" on Web Sites	6

[1.](#) Introduction

Many Web sites have a "safe" mode, to assist those who don't want to be exposed (or have their children exposed) to "objectionable" content.

However, that goal is often difficult to achieve, because of the need to go to each Web site in turn, navigate to the appropriate page (possibly creating an account along the way) to get a cookie [[RFC6265](#)] set in the browser, for each browser on every device used.

If this desire is proactively advertised by the user agent, things become much simpler. A user agent that supports doing so (whether it be an individual browser, or through an Operating System HTTP library) need only be configured once to assure that the preference is advertised to a set of sites, or even all sites.

This specification defines how to declare this desire in requests as a HTTP Preference [[RFC7240](#)].

Note that this specification does not precisely define what "safe" is; rather, it is interpreted within the scope of each Web site that chooses to act upon this information. Furthermore, sending "safe" does not guarantee that the Web site will use it.

That said, the intent of "safe" is to allow end users (or those acting on their behalf) to express a desire to avoid content that is considered "objectionable" within the cultural context of that site; usually (but not always) content that is unsuitable for minors. The "safe" preference ought not be used for other purposes.

It is also important to note that the "safe" preference is not a reliable indicator that the end user is a child; other users might have a desire for unobjectionable content, and some children might browse without the preference being set.

Nottingham

Expires August 17, 2015

[Page 2]

Simply put, it is a statement by (or on behalf of) the end user to the effect "If your site has a 'safe' setting, this user is hereby opting into that, according to your definition of the term."

1.1. 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 [\[RFC2119\]](#).

2. The "safe" Preference

When present in a request, the "safe" preference indicates that the content which is not objectionable is preferred, according to the server's definition of the concept.

For example, a request that includes the "safe" preference:

```
GET /foo.html HTTP/1.1
Host: www.example.org
User-Agent: ExampleBrowser/1.0
Prefer: safe
```

User agents SHOULD include the "safe" preference in all requests, to ensure that the preference is available to the applicable resources. Note that the resources which "safe" is sent to is potentially configurable; see [Appendix B](#) for more information.

Additionally, other clients MAY insert it; e.g., an operating system might choose to insert the preference in requests based upon system-wide configuration.

Origin servers that utilize the "safe" preference ought to document that they do so, along with the criteria that they use to denote objectionable content. If a server has more fine-grained degrees of "safety", it SHOULD select a reasonable default to use, and document that; it MAY use additional mechanisms (e.g., cookies [\[RFC6265\]](#)) to fine-tune.

A response corresponding to the request above might have headers that look like this:

```
HTTP/1.1 200 OK
Transfer-Encoding: chunked
Content-Type: text/html
Server: ExampleServer/2.0
Vary: Prefer
```


Note that the Vary response header needs to be sent if cacheable responses associated with the resource might change depending on the value of the "Prefer" header. This is not only true for those responses that are "safe", but also the default "unsafe" response.

See [\[RFC7234\] Section 4.1](#) for more information the interaction between Vary and Web caching.

See [Appendix C](#) for additional advice specific to Web servers wishing to use "safe".

3. Implementation Status

Note to RFC Editor: Please remove this section before publication.

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft. The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

- o Microsoft Internet Explorer - see <http://support.microsoft.com/kb/2980016>
- o Microsoft Bing
- o Mozilla Firefox - see https://bugzilla.mozilla.org/show_bug.cgi?id=995070
- o Cisco - see <http://blogs.cisco.com/security/filtering-explicit-content>
- o YouTube - based upon testing the live site (not formally announced)

4. Security Considerations

The "safe" preference is not a secure mechanism; it can be inserted or removed by intermediaries with access to the request stream (e.g. for "http://" URLs). Its presence reveals limited information about the user, which may be of small assistance in "fingerprinting" the user.

By its nature, including "safe" in requests does not assure that all content will actually be safe; it is only when servers elect to honor it that content might be "safe".

Even then, a malicious server might adapt content so that it is even less "safe" (by some definition of the word). As such, this mechanism on its own is not enough to assure that only "safe" content is seen; those who wish to ensure that will need to combine its use with other techniques (e.g., content filtering).

Furthermore, the server and user may have differing ideas regarding the semantics of "safe." As such, the "safety" of the user's experience when browsing from site to site might (and probably will) change.

5. IANA Considerations

This specification registers the "safe" preference [[RFC7240](#)]:

- o Preference: safe
- o Value: (no value)
- o Description: Indicates that "safe" / "unobjectionable" content is preferred.
- o Reference: (this document)
- o Notes:

6. References

6.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC7240] Snell, J., "Prefer Header for HTTP", [RFC 7240](#), June 2014.

6.2. Informative References

- [RFC6265] Barth, A., "HTTP State Management Mechanism", [RFC 6265](#), April 2011.
- [RFC7234] Fielding, R., Nottingham, M., and J. Reschke, "Hypertext Transfer Protocol (HTTP/1.1): Caching", [RFC 7234](#), June 2014.

[Appendix A.](#) Acknowledgements

Thanks to Alissa Cooper, Ilya Grigorik, Emma Llanso, Jeff Hughes, Lorrie Cranor, Doug Turner and Dave Crocker for their comments.

[Appendix B.](#) Setting "safe" from Web Browsers

As discussed in [Section 2](#), there are many possible ways for the "safe" preference to be generated. One possibility is for a Web browser to allow its users to configure the preference to be sent.

When doing so, it is important not to misrepresent the preference as binding to Web sites. For example, an appropriate setting might be a checkbox with wording such as:

☐ Request "safe" content from Web sites

... along with further information available upon request (e.g., from a "help" system).

Browsers might also allow the "safe" preference to be "locked" - that is, prevent modification without administrative access, or a passcode.

Note that this specification does not constrain browsers to send "safe" on all requests, although that is one possible implementation; e.g., an alternate implementation strategy would be to allow a blacklist (of sites that "safe" is not sent to).

[Appendix C.](#) Supporting "safe" on Web Sites

Web sites that allow configuration of a "safe" mode (for example, using a cookie) can add support for the "safe" preference incrementally; since the preference will not be supported by all clients immediately, it is necessary to have another way to configure it.

When honoring the safe preference, it is important that it not be possible to disable it through the Web site's interface, since "safe" may be configured and locked down by the browser's administrator (e.g., a parent). If the site has configuration (e.g., stored user preferences) and the safe preference is received in a request, the "safer" interpretation is always used.

If the user expresses a wish to disable "safe" mode, the site should remind them that the safe preference is being sent, and ask them to consult their administrator (since "safe" might be set by a locked-down Operating System configuration).

As explained in [Section 2](#), responses that change based upon the presence of the "safe" preference need to either carry the "Vary: Prefer" response header field, or be uncacheable by shared caches (e.g., with a "Cache-Control: private" response header field). This is to avoid an unsafe cached response being served to a client that prefers safe content (or vice versa).

Author's Address

Mark Nottingham

EMail: mnot@mnot.net

URI: <http://www.mnot.net/>