



IETF HTTP WG 2020-10 Interim - SEARCH Method

[Julian Reschke](#), greenbytes

Slides below are mostly from IETF 93 (Prague, 2015).

[Meeting minutes](#) show that Call for Adoption (on the mailing list) was planned (but did not happen).

Why?

One of the most FAQs on StackOverflow is: can I send a GET request with a payload?

We should have material that explains why GET with body doesn't work, and what the alternatives are.

Proposal:

Explain the situation:

- Bookmarkability/Cacheability
- Drawbacks of using GET: URIs might leak more frequently than payloads (log files, Referer), some components might fail for long URIs
- Drawbacks of using POST: not safe, thus not repeatable without knowing the semantics of the request
- Explore Content-Location, making the GET-table resource discoverable
- Can URI templates help?

Alternatives to GET and POST

- Use an HTTP method that is defined to support a request payload **and** is safe.
- The method registry already contains three candidates: PROPFIND, REPORT, and SEARCH; all of which defined for WebDAV.
- There is existing code out there which knows about them being safe, so it makes sense to use one of these. It also avoids using yet another method name.
- The most generic of these is SEARCH; we could un-tangle it from WebDAV (without breaking existing uses) by allowing any media type as payload (as in PATCH), and open up the response format as well.

Specifically...:

- Make it as simple as possible.
- Format discovery using Accept-Search response header field (mirrors PATCH).
- Discuss concrete formats in separate specs.
- Explore ways to make the response to SEARCH GET-table and to leverage URI templates so clients can directly construct GET requests once they know about the URI format.

Further reading:

- [draft-snell-search-method-02](#)
- [RFC 5323: Web Distributed Authoring and Versioning \(WebDAV\) SEARCH](#)
- [Latest discussion thread on WG mailing list](#)