Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 23 January 2026

M. Nottingham
S. Newton
Cloudflare

22 July 2025

Requirements for Paid Web Crawling
draft-nottingham-paid-crawl-reqs-00

Abstract

This document suggests requirements (and non-requirements) for paid Web crawling protocols.

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 23 January 2026.

Copyright Notice

Table of Contents

1. Introduction

Automated web clients, or "crawlers," have increasingly dominated website traffic, leading to increased operational costs and new technical and economic risks.

Historically, websites have borne these costs and risks because they believed they received value in return. For instance, crawling to build web search indices exposed sites to increased "referral" traffic when search engine users clicked links to those sites.

However, this balance has been disrupted by an increase in web traffic without any corresponding benefit to websites. Crawling to train Large Language Models ("AI") not only burdens site infrastructure but also creates value for the LLM vendor without compensation to the content owner.

An Internet protocol to facilitate payments from crawlers to websites could help address this imbalance. This document outlines the use case in Section 1.1, specifies requirements in Section 2, and identifies non-requirements in Section 3.

1.1. The Paid Web Crawling Use Case

A Web site "S" wants to be financially compensated for a Web client "C"'s access to its resources. This might be facilitated by a payment processor, "P".

For purposes of this use case, we assume:

* C is _not_ a Web browser with a human behind it; it is a machine-driven process that is collecting Web content for some other purpose (colloquially, "crawling" the Web). Note that that process might (or might not) use a "headless" browser as part of its operation.

* There are a diverse set of C in the world, but the total set of C that a site will interact with is reasonably bounded (i.e., there will not be thousands of C accessing a given site with this protocol, but there may be twenty or more).

* S has some means of cryptographically identifying C. See https://datatracker.ietf.org/wg/webbotauth/about/ (https://datatracker.ietf.org/wg/webbotauth/about/).

Note that this use case is not uniformly applied to all Web crawlers; the intent is not to preclude or require payment for all crawlers, but instead to address situations where there is an economic imbalance.

1.2. 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. Requirements

The following sections propose requirements for a protocol that facilitates payment by crawlers to Web sites.

2.1. Avoid Centralization

A crawl payment protocol MUST NOT have a single or constrained number of "choke point" or "gatekeeper" roles. It MUST be possible for new payment processors to be introduced into the ecosystem with reasonable effort. In particular, attention needs to be paid to mitigating factors such as network effects.

Furthermore, a crawl payment protocol SHOULD NOT have secondary effects that encourage centralization in either clients (e.g., allowing advantages to accrue to a small number of well-known crawlers) or servers (e.g., creating significant barriers to deploying new Web sites that compete with well-known ones). Where they are unavoidable, these effects SHOULD be mitigated if at all possible.

Similarly, a crawl payment protocol MUST NOT "bundle" other capabilities unless absolutely necessary to achieve its goals. For example, it SHOULD NOT require the payment processor to be co-located with the server, or with the party providing access control to the server.

See [CENTRALIZATION] for further discussion.

2.2. Lower Deployment Costs

A crawl payment protocol MUST be reasonable to deploy in a variety of systems. In particular, it SHOULD NOT incur significant processing, network, or storage overheads on the servers that wish to require payment. It SHOULD be compatible with common techniques for efficient Web sites, such as caching, serving from a filesystem, and in particular SHOULD NOT incur significant per-request overhead, unless absolutely necessary to meet the goals of the protocol.

It is acknowledged that "significant" is in the eye of the beholder, and can vary based upon the resources available to a system. Here, the intent is to allow deployment on a diversity of systems, thereby helping to avoid the centralization risks described in Section 2.1. Thus, a successful crawl payment protocol SHOULD be deployable on a reasonable variety of systems that include at least one maintained by a single person on commodity hardware, but might not reach to some more specialised systems, such as a low-power embedded server.

2.3. Be Extensible

One of the core requirements for Internet protocols is the ability to evolve -- to incorporate new use cases as well as changes in their context and use. Because the Internet is a distributed system, we cannot call a "flag day" where everyone changes at once; instead, changes are accommodated through explicit extensibility mechanisms. See [EXTENSIBILITY] for more discussion.

Therefore, a crawl payment protocol MUST allow a variety of payment schemes to be used with it, and MUST allow introduction of new capabilities.

Particular attention will need to be paid to the deployability of such extensions. If a small set of payment schemes is deployed, it may be difficult for sites to introduce a new one without protocol support (e.g., fallback mechanisms).

2.4. Minimise Information Disclosure

A crawl payment protocol SHOULD NOT expose more information about either party than is necessary to complete the payment. Note that legal requirements in some jurisdictions and payment regimes may require exposure of such information, but it SHOULD be limited to that which is required.

Furthermore, a crawl payment protocol MUST NOT expose additional information about the parties publicly.

This requirement extends to the terms of the payment itself: some parties may not wish to make the amount they are paying or being paid for crawling public information.

See [PRIVACY] for more considerations regarding privacy in protocol design.

2.5. Allow Granularity

A crawl payment protocol SHOULD allow sites to have separate payment agreements for different sets of content on them. This reflects the nature of content: some of it is more valuable or more expensive to produce.

Note that this is not an absolute requirement: granularity often comes at a cost of complexity and protocol "chattiness," which are in tension with other requirements.

2.6. Facilitate Negotiation

A crawl payment protocol SHOULD allow the parties to negotiate over time, so that they can converge on a payment that is agreeable to both of them. However, because negotiation adds complexity to the protocol (and therefore implementation and deployment burden), it SHOULD be optional to use for both parties; i.e. either party could make a "take it or leave it" offer.

Likewise, a crawl payment protocol MAY consider providing some level of price transparency either directly or indirectly (e.g., through intermediarie), provided that the privacy requirements in Section 2.4 are met.

2.7. Allow Enforcement

A crawl payment protocol SHOULD allow intermediaries acting on behalf of the origin server to verify payment status, so that they can impose policy.

3. Non-Requirements

To clarify the scope of work, the following items are considered as NOT being requirements for a successful crawl payment protocol.

Note that in each case, this does not preclude a successful protocol from accommodating the non-requirement, or require the protocol to preclude that end: it only implies that the non-requirement is not a design goal that the effort will actively seek.

3.1. Standalone Deployment

While we wish to avoid centralization (see Section 2.1), it is not a requirement to facilitate full deployment of the protocol exclusively on a single Web server, without external dependencies.

This non-requirement reflects the nature of payment systems, which typically use intermediaries to provide useful services such as chargebacks, reputation management, and compliance with legal requirements.

The implication is that where payment intermediaries are used in the protocol, they should be as interchangeable as possible, to promote an ecosystem whereby both servers and crawlers have choices regarding which intermediaries they support.

3.2. Real-Time Operation

It is not a requirement for the protocol to facilitate immediate payment at request time, though the protocol may allow for this. Crawlers are not like Web browsers: they are long-running processes that aren’t constrained by the responsiveness requirements of human users, and can reconcile asynchronous operations.

3.3. Controlling Content

It is not a requirement to provide a technical means of controlling the use of content once it has been crawled; this is not a Digital Rights Management scheme.

3.4. Preventing Bad Faith

Some crawlers will attempt to crawl without using a payment protocol (e.g., by masquerading as browsers). It is not a requirement of a crawl payment protocol to prevent such misuse. Instead, we expect other interventions -- including blocking of misbehaving crawlers -- to disincent such behaviour.

Some crawlers might even use contents for purposes other than what they negotiate. Likewise, some sites might renege on their agreements and refuse access to content that a crawler has paid for. It is not a requirement to technically prevent these situations. We expect such cases to be addressed by other mechanisms, such as legal intervention.

4. IANA Considerations

This document has no tasks for IANA.

5. Security Considerations

Payment mechanisms for Web crawling undoubtedly have security implications and considerations, but beyond the aspects captured above, it is premature to characterise their nature.

6. References

6.1. Normative References

[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>.

6.2. Informative References

[CENTRALIZATION] Nottingham, M., "Centralization, Decentralization, and Internet Standards", RFC 9518, DOI 10.17487/RFC9518, December 2023, <https://www.rfc-editor.org/rfc/rfc9518>.

[EXTENSIBILITY]
Thomson, M. and T. Pauly, "Long-Term Viability of Protocol Extension Mechanisms", RFC 9170, DOI 10.17487/RFC9170, December 2021, <https://www.rfc-editor.org/rfc/rfc9170>.

[PRIVACY] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, "Privacy Considerations for Internet Protocols", RFC 6973, DOI 10.17487/RFC6973, July 2013, <https://www.rfc-editor.org/rfc/rfc6973>.

Authors' Addresses

Mark Nottingham
Cloudflare
Melbourne
Australia
Email: mnot@mnot.net
URI: https://www.mnot.net/

Simon Newton
Cloudflare
Cambridge
United Kingdom
Email: rfc@simonnewton.com