Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 9 June 2026

M. Nottingham

6 December 2025

Use Cases for Cryptographic Authentication of Web Bots
draft-nottingham-webbotauth-use-cases-00

Abstract

This draft outlines use cases for cryptographic authentication for bot clients on the Web.

About This Document

This note is to be removed before publishing as an RFC.

Status information for this document may be found at https://datatracker.ietf.org/doc/draft-nottingham-webbotauth-use-cases/.

information can be found at https://mnot.github.io/I-D/.

Source for this draft and an issue tracker can be found at https://github.com/mnot/I-D/labels/webbotauth-use-cases.

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 9 June 2026.

Copyright Notice

Table of Contents

1. Introduction

The Web Bot Auth (WebBotAuth) Working Group has been chartered to "standardize methods for cryptographically authenticating automated clients and providing additional information about their operators to Web sites."

Initial discussions in the group have revealed some disagreement about the scope and intent of the work. Section 2 explores the use cases for cryptographic authentication of non-browser clients, to help inform those discussions. Section 3 suggests some further questions for consideration.

2. Web Site Use Cases

This section explores use cases that Web sites might have for authenticating bots, including a discussion of any current mechanisms that they use to meet the use case and how cryptographic authentication might help.

Because there is some question about the "additional information" facility in the charter, each use case also assesses whether it's necessary to identify a real-world entity associated with the bot to meet the use case (since that is the most common use of such a facility).

Each use case also summarises how controversial addressing it is perceived to be.

This draft does not take a position on whether all of the use cases should be addressed by the group. Potential alternative solutions to the implied requirements are also not considered here.

2.1. Mitigating Volumetric Abuse by Bots

Some bots make requests at rates that cause operational issues for Web sites. This may be intentional (e.g., traffic from "botnets" and other attacks) or unintentional (due to overly simple or inconsiderate implementation). It appears that both the number of such bots and the rate at which they make requests are increasing.

While sites can take measures to mitigate the impact of this traffic (e.g., caching), these are only partially effective; some resources are uncacheable, and generating representations of some HTTP resources can incur much higher costs. In general, serving such great volumes of traffic can consume significant resources, in terms of both infrastructure and bandwidth.

Currently, a site that experiences such traffic most often blocks unwelcome clients by IP address. This has the effect of blocking other uses of that IP address, both at that time and into the indefinite future. It also offers little recourse for incorrectly blocked clients, since they have no information about why they were blocked or what they should do about it.

Cryptographic authentication of bots could allow a site that experiences abusive traffic to allow those automated clients that are authenticated and well-behaved, blocking any from the remaining who make more requests than a Web browser conceivably could. Unlike IP-based blocking, cryptographic authentication is specific to the bot itself, limiting harmful effects to other users of the IP address (either concurrently or in the future). Because it is performed at the application layer, it gives the opportunity to convey additional information to the client (e.g., in a HTTP 403 or 429 response).

This use case does not require identifying a specific bot or associating it with a real-world entity, provided that the site considers abusiveness a feature of behaviour, not identity. It also does not require discriminating between bots and non-bot users; only the problematic behaviour is targeted.

Addressing this use case does not appear to be overly controversial, because it is widely recognised that a site needs to operate with reasonable efficiency to provide both its operators and its users a benefit.

2.2. Controlling Access by Bots

Some sites wish to make access by bots to the resources they provided to browsers conditional upon the identity or features of the bot. This might be for a variety of reasons; they may wish to:

* Only allow access by bots on an allow list;

* Disallow access to bots on an explicit deny list;

* Condition access upon meeting some criteria (e.g., non-profit, certification by a third party);

* Condition access upon participation in some scheme or protocol (e.g., payment for access);

Note that the first two imply some notion of bots being tied to a real-world identity, whereas the remaining do not necessarily require it.

Currently, sites most often use a combination of the Robots Exclusion Protocol (including robots.txt) and IP address blocking to control access by bots.

The Robots Exclusion Protocol provides a means for sites to communicate preferences to bots about their behaviour. Although this is a useful and sometimes necessary function, it does not allow for enforcement of those preferences.

Enforcement is achieved primarily through blocking non-conforming clients. The limitations of IP address blocking are discussed in Section 2.1.

Cryptographic authentication of bots could allow for greater certainty in access control than is possible using IP addresses. It would also allow for more definite tying of roles to actors in protocols (e.g., in the Robots Exclusion Protocol and in any potential payment protocol).

This use case has been disputed. While blocking certain bots by IP address is widespread in practice, concerns have been expressed that standardising an authentication mechanism for bots might result in a Web where all bots might need to authenticate, leading to increased difficulty in introducing new bots. In some markets, this outcome could create pressure towards centralisation, due to heightened barriers to entry.

Another controversy is that giving sites a more fine-grained capability to block bots is a change in the balance of power on the Web. Some perceive that as justified, given factors like the introduction of AI and what they perceive as an onslaught of bot traffic. Others see it as an overreach that may impinge upon users' ability to consume content as they desire -- for example, using accessibility or agentic tools.

Finally, some see bots as a way of keeping powerful sites in check, and therefore measures to curtail their activity is portrayed as concentrating that power. However, it should be noted that there are also powerful bots that can be seen to have disproportionate power over sites, and so there is not necessarily a clear bias here.

2.3. Providing Different Content to Bots

Somes sites may wish to tailor the content they serve to bots (either selectively or overall), as compared to that they serve to browsers. In some cases, a site might wish to augment the information that they provide to a trusted bot. Conversely, a site might wish to reduce or modify the information that they provide to a bot that they do not trust.

Current practice is difficult to ascertain, but anecdotal evidence suggests that the latter case is more common than the former. For example, some sites do not wish for information that they consider to be commercially sensitive -- e.g., prices -- to be available to bots. In both cases, IP addresses and similar heuristics are used.

Cryptographic authentication of bots could enable such discrimination in a manner much more reliable than using other techniques (such as IP addresses), provided that it also allowed identification of individual bots.

In most cases, this use requires identifying a specific bot and associating it with a real-world entity (although there are exceptions, such as sites which want to treat all bots equally, or cases where it's possible to group bots without identifying specific ones).

This use case is likely to be controversial in cases where the modifications are not consensual. Some espouse a site's right to control its own speech depending upon the audience it is speaking to, whereas others are concerned by the lack of transparency that might result -- particularly from powerful sites. Note, however, that a bot that cannot be distinguished from a typical browser is still likely to be able to operate for such purposes.

2.4. Auditing Bot Behaviour

Some sites may wish to understand how bots use them in detail. In particular, they might want to verify that a particular bot adheres to the preferences stated in robots.txt, or that they conform to some other protocol. They might also wish to have reliable metrics for how a bot behaves in terms of number of requests, timing of requests, and so forth to ascertain the bot's behaviour; this information might feed into other use cases, or be used independently.

Currently, this use case is met through use of heuristics of information like IP address.

Cryptographic authentication of bots could allow more accurate and reliable auditing of bot behaviour, due to the greater confidence it provides. It does not necessarily require identifying a specific bot or associating it with a real-world entity, but some (many?) of the downstream uses of the audit data may.

This use case does not appear controversial, because bots being accountable for their behaviour is broadly seen as a reasonable goal.

2.5. Classifying Traffic

Many sites make efforts to understand how browsers interact with them, so as to improve their services. This might be at the connection level (e.g., HTTP, TCP, and QUIC statistics), or it might be gathered in-browser (Real User Monitoring or RUM).

When doing so, it is important for them to be able to distinguish between their target audience (people using browsers) and bots; if they cannot, the bot traffic will make the insights they gain less useful (or even useless).

Currently, sites that perform such tasks use a variety of heuristics to identify and exclude bots from such measures. This is only partially effective; bots are increasingly difficult to classify, particularly as using ’headless browsers’ becomes a norm for crawlers.

Cryptographic authentication could be an aid to classification if it becomes a norm for bots to use it. It does not require identifying specific bots or associating them with real-world entities unless finer granularity of classification than "bot vs not" is desired. However, sites that wish to exclude non-human clients from their measurements would still need to use heuristics for bots that do not comply with the norm.

Addressing this use case does not appear to be controversial, because an understanding of the nature of traffic that a site receives is important to its operation (provided that no personal information is involved and no tracking capability is introduced).

2.6. Site Services

Many sites use third-party tools to analyse, monitor, and provide their services. For example, health check services allow sites to understand their uptimes and receive notifications when there is a problem. Content Delivery Networks need to identify themselves to back-end origin servers.

Currently, such services use a variety of means of authentication, including IP address allow lists, "magic" header fields, and ad hoc use of existing cryptographic mechanisms.

Site services often have higher requirements for reliability and security. A site might not wish to grant access to a vulnerability scanner solely based upon its IP address, for example. Likewise, a health check needs to reliably bypass Web Application Firewalls to perform its function.

A standard cryptographic authentication mechanism for bots could improve security and reliability, and provide greater certainty in the face of operational changes (e.g., changes to the bot’s IP addresses). This use case requires bot identity to be tied to authentication.

Addressing this use case does not appear to be controversial. However, it is not clear whether these use cases are within the scope of the Working Group’s charter.

3. Next Steps

This section suggests questions for further investigation and discussion.

1. What are the qualitative differences between current practice (e.g. ad hoc blocking by IP address) and cryptographic authentication of bots?

2. User authentication is widespread and standards-supported on the Web; what makes bot authentication different?

3. What levers do we have to mitigate the harms associated with an emerging default of requiring authentication for bots? Does cryptographic authentication enhance or confound such efforts (as opposed to IP address blocking)?

4. Would a cryptographic authentication scheme that does not allow association with real-world entities provide enough value to meet interesting use cases? If so, would the charter prohibition on "[t]racking or assigning reputation to particular bots" need to change?

5. What is the threshold for being considered a bot? E.g., is request rate important? Associating with a specific human user in time and/or space?

6. Are the resource requirements for cryptographic authentication reasonable for these use cases for all types of sites? At the meeting, it was asserted that it would disproportionately advantage already well-resourced entities.

7. What use cases should the group address and not address? Why?

8. Are there alternative approaches to addressing some or all of these use cases? What properties do they have?

4. IANA Considerations

This draft has no actions for IANA.

5. Security Considerations

Undoubtedly there are security considerations to a cryptographic authentication protocol, but they will be encountered and dealt with later than what’s in scope for this draft.

Appendix A. Bot Differences

This section enumerates some of the ways that bots can differ.

A.1. Scope

Bots have different scopes of activity:

* Some crawl the entire Web
* Some target a specific subset of the Web (e.g., by geography, language, industry)
* Some target specific sites or resources on sites (e.g., link checkers, linters)

A.2. Relationship

Bots have different relationships with sites:

* Some actively attempt to appear as Web browsers, so as to have the same relationship as an end user
* Some do not hide their nature as bots but do not have any pre-existing relationship with the site
* Some are implicitly or explicitly authorised by the site (e.g., through an advertised API)
* Some have a pre-existing relationship with the site (e.g., monitoring and other site services)

A.3. Reputation

Bots have different reputations in the larger community, which can change how they are perceived by sites:

* Some are well and widely-known (e.g., search engine crawlers, archivers)
* Some are relatively unknown (e.g., due to low traffic or recent introduction)
* Some are purposefully anonymous (e.g., price checkers, most malicious bots)

A.4. Agency

Bots act with different relationships to their operator(s):

* Some are explicitly and exclusively associated with an end user (e.g., "agentic" bots)

* Some are acting on behalf of a group of end users

* Some are acting on behalf of another entity (e.g., corporation, government, civil society organisation)

* Some serve multiple constituencies

Author’s Address

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