No Request With a Known Crawler User-Agent Reached llms.txt: Fifteen Days of Logs at One Origin Server

Evgenii Arsentev, PhD

ORCID: 0000-0002-9120-7298 · ARSENTEV.AI <https://arsentev.ai>

Date: 28 September 2026 · **Version:** 1.0 (technical report)

DOI: 10.5281/zenodo.23018429 · **License:** CC BY 4.0

Related: Internet-Draft draft-arsentev-llm-context-discovery-00 [4] (IETF Datatracker, September 2026). Section 6 of this report corrects the measurement figures printed in that draft.

Abstract

Many publishers now place a plain-text summary of their site at `/llms.txt` (and a longer version at `/llms-full.txt`) in the hope that language-model crawlers will read it. No standard tells a crawler that such a file exists. I measured whether crawlers fetched it anyway. The data are the complete nginx access logs of one origin server for 28 August to 11 September 2026 (15 days, 345,808 parsed requests). A request was attributed to a crawler when its User-Agent contained one of 22 tokens published by search and AI operators. Crawler tokens accounted for 46,155 requests from 20 distinct tokens (Table 2). Not one of these requests was for `/llms.txt` or `/llms-full.txt`. From 9 to 11 September (the files were first served, at the latest, on 9 September at 23:07 UTC), crawlers made 15,906 requests, including 698 for `/robots.txt` (from 11 tokens) and 282 for `/sitemap.xml`, and again none for the context files. All 67 requests for the context files in the window came from clients without a crawler token (53 of them command-line HTTP clients). No crawler token appeared on any path ending in `llms.txt` either, including `/.well-known/llms.txt`. The observation is small and local: one origin, a two-day exposure after deployment, and self-asserted crawler identity. It says that in this setting the context files were not requested by any client presenting a listed crawler token at the origin, while `robots.txt`, which crawlers fetch by protocol, and the sitemap were fetched hundreds of times. The report also corrects the figures published in Internet-Draft -00, whose totals match the log cut at about 12:09 UTC on the last day.

1. Question

The `llms.txt` proposal [1] asks site owners to publish a Markdown summary at a fixed path. Unlike `robots.txt` (RFC 9309 [2]) or an XML sitemap [3], the path is not referenced by any protocol that crawlers already implement. A crawler that has not been specifically programmed to request the path has no way to learn that the file exists, unless a page or a sitemap lists it. The question here is narrow: over a fixed window, did any request carrying a known crawler User-Agent retrieve the context files at a server that served them? And, as a comparison inside the same traffic, how often did the same crawlers retrieve robots.txt, which they fetch by protocol, and the sitemap?

2. Setting

The origin server is a single nginx instance behind the Cloudflare CDN. During the window it served three public hostnames: `arsentev.ai` and `ru.arsentev.ai` (one personal research site in English and Russian), and `dad4kids.com` (a children's educational games site by the same publisher). All three wrote to one access log in the nginx *combined* format, which does not record the requested hostname. Counts in this report are therefore pooled over the three hostnames. The context files existed only on the two `arsentev.ai` hostnames; `dad4kids.com/llms.txt` returns 404 (checked 28 September 2026). The files were generated from the same source as the HTML pages. `robots.txt` has no field for such a file. A sitemap could list it as an ordinary URL, and an HTML page could link to it. Neither the sitemap contents nor page links during the window were recorded, so this report does not know whether the files were linked. (On 28 September, `robots.txt` carried `Sitemap:` lines and the sitemap listed the context files; that is not evidence about the window.) All three hostnames resolved to Cloudflare addresses when checked on 28 September. The log records only requests that reached the origin. Requests answered from the CDN cache never appear in it: on 28 September `robots.txt` was cached at the edge (`cf-cache-status: EXPIRED`), so robots counts here are lower bounds, while `/llms.txt` was not (`DYNAMIC`). The cache configuration during the window was not recorded.

3. Method

3.1 Data

Raw data are the 15 rotated files of `/var/log/nginx/access.log`, copied on 12 September 2026 before rotation would have removed the oldest day (SHA-256 checksums are released with the data). The first line is dated 28 August 2026 00:55 UTC and the last 11 September 2026 23:16 UTC. The files contain 346,150 lines. 342 lines did not match the parsing expression and were excluded, which leaves 345,808 requests, all inside the window.

3.2 Crawler attribution

A request is attributed to a crawler if its User-Agent header contains, case-insensitively, one of these 22 tokens (checked in this order; the first match wins): GPTBot, ChatGPT-User, OAI-SearchBot, Claude-User, Claude-SearchBot, ClaudeBot, anthropic-ai, Perplexity-User, PerplexityBot, Google-Extended, Googlebot, bingbot, Applebot, Amazonbot, Bytespider, CCBot, meta-externalagent, FacebookBot, YandexBot, DuckDuckBot, cohere-ai, MistralAI-User. Twenty of the 22 tokens occurred; anthropic-ai and FacebookBot did not. The order puts specific tokens first. That matters because operators' own User-Agents embed a second token: 11 lines matched two tokens, all of them in the operators' documented formats (Claude-SearchBot and Claude-User strings contain `claudebot@anthropic.com`, Perplexity-User strings contain `perplexity.ai/perplexitybot`). Each line is counted once, under the specific token. Googlebot includes 471 requests from Googlebot-Image. The list mixes training crawlers, search-index crawlers and user-triggered fetchers, because the question is whether any automated consumer found the file, whatever its purpose.

3.3 Paths and periods

A request counts as a context-file retrieval if its path, with the query string removed, is exactly `/llms.txt` or `/llms-full.txt`. The same rule is used for `/robots.txt` and `/sitemap.xml`. Any response status is counted, since the question is whether the path was requested at all. The *post-deployment period* is defined in two ways. The day-based definition (used in the released script and in draft -00) is 9 to 11 September inclusive. The stricter definition starts at the first HTTP 200 response for `/llms.txt` in the log, 2026-09-09 23:07:52 UTC,

and runs to the end of the window, about 48 hours. The first 200 for `/llms-full.txt` came later, on 10 September at 20:21 UTC. The actual deployment times were not recorded; first responses bound them from above. As a looser check, I also counted requests for any path ending in `llms.txt` or `llms-full.txt`.

3.4 Identity check

User-Agent strings are self-asserted. Operators publish address ranges for some crawlers, but address checks were not possible for most of this traffic: nginx logged the connecting address, and for requests that came through the CDN that address is a Cloudflare edge node. I split crawler requests into those whose logged address lies in Cloudflare's published ranges and those that reached the origin directly. For the direct requests, I tested the logged address against the ranges published by Google [5], OpenAI [6], Microsoft (Bing) [8], Apple [9], Perplexity [10] and DuckDuckGo [11]; Cloudflare's ranges are [7]. The range files were downloaded on 28 September 2026, 17 days after the window closed.

3.5 Reproducibility

Two scripts produce every number in this report from the raw logs. `analyze.py` gives the headline counts (Table 1) and the day-by-token aggregate. `supplement.py` gives the per-token table, daily totals, context-file requests by client class, the CDN split and the stricter post-deployment counts; it asserts that its totals equal those of `analyze.py`. Setting `CUTOFF=2026-09-11T12:10:00` for `analyze.py` gives the truncated-log variant in Section 6, and `cut_search.py` finds the cut times discussed there. All three scripts use only the Python standard library. The released files are aggregates. They contain no IP addresses, referrers or per-visitor rows, so the raw logs are not released. Because of that, a third party can check the internal consistency of the aggregates and rerun the scripts on their own logs, but cannot re-derive these figures from the released files alone.

4. Results

Quantity	Full window (28 Aug to 11 Sep)	9 to 11 Sep (day-based)	From first 200 for /llms.txt
Requests, all clients	345,808	106,712	n/a
Requests with a crawler token	46,155	15,906	12,817
Distinct crawler tokens seen	20	18	18
Crawler requests for /robots.txt	2,135	698	511
Distinct tokens requesting /robots.txt	n/a	11	11
Crawler requests for /sitemap.xml	n/a	282	191
Crawler requests for /llms.txt or /llms-full.txt	**0**	**0**	**0**

Table 1. Headline counts. "n/a" marks quantities the scripts do not compute for that period. The full-window /robots.txt figure is the sum of response statuses in Section 4.3.

4.1 By crawler token

Token	Requests, full window	Requests, 9 to 11 Sep	/robots.txt, 9 to 11 Sep	/sitemap.xml, 9 to 11 Sep	Context files
Googlebot	8,759	4,262	340	4	0
GPTBot	4,674	3,032	0	4	0
Claude-SearchBot	4,103	460	46	170	0
YandexBot	3,432	2,164	102	1	0
Applebot	2,882	670	22	0	0
PerplexityBot	2,871	775	4	0	0
Claude-User	2,867	380	5	0	0
bingbot	2,503	581	13	23	0
Amazonbot	2,221	704	2	0	0
ChatGPT-User	2,212	389	0	0	0
OAI-SearchBot	2,182	352	32	0	0
ClaudeBot	2,007	843	128	79	0
Perplexity-User	1,818	259	0	0	0
meta-externalagent	1,489	740	0	1	0
Bytespider	908	120	4	0	0
Google-Extended	863	118	0	0	0
DuckDuckBot	229	44	0	0	0
CCBot	124	13	0	0	0
cohere-ai	7	0	0	0	0
MistralAI-User	4	0	0	0	0
Total	**46,155**	**15,906**	**698**	**282**	**0**

Table 2. Requests presenting each crawler token (not verified operator activity, see Section 5). "Context files" is /llms.txt plus /llms-full.txt over the full window.

4.2 Who did request the context files

The log holds 67 requests for the two context-file paths: 51 answered with 200 and 16 with 404. Ten of the 404s came before deployment. The other six came after it; since the log has no hostname, they may be requests to `dad4kids.com`, which has no such file. None of the 67 carried a crawler token. I grouped the clients by User-Agent into four classes (Table 3). All 51 responses with status 200 went to `curl`; 35 of them fell on 11 September. Draft -00 attributed these requests to the publisher's own checks of the deployment. The logged addresses cannot confirm that, so this report makes no claim about who ran those clients. The single self-declared bot was an unrelated crawler that received a 404 on 8 September, before deployment. Two more requests, for `/.well-known/llms.txt`, came from clients without a crawler token and received 404.

Client class (by User-Agent)	Full window	9 to 11 Sep
Command-line or library HTTP client (curl and similar)	53	53
Browser-like (Mozilla/... with no bot marker)	13	4
Other self-declared bot, not on the token list	1	0
Crawler token from the list	0	0
Total	**67**	**57**

Table 3. Requests for /llms.txt and /llms-full.txt by client class.

4.3 Other checks

Over the full window, crawler requests for `/robots.txt` received 200 in 1,892 cases, a 301 redirect (to the canonical host or scheme) in 242 and a 502 in one. 2,238 crawler requests (4.8%) went to paths matching `^/(en|ru)/game/|^/games/|^/game/`, which by the author's knowledge of the two sites' routes exist only on `dad4kids.com`. This is a lower bound on the second site's share, since paths common to both sites cannot be told apart. Daily totals are in Table 4. The total for 11 September (57,679) is higher than on other days. I did not look into the cause. Crawler-attributed requests that day (5,817) were lower than on 10 September (6,949).

Day	All	Crawler	Day	All	Crawler	Day	All	Crawler
28 Aug	9,994	1,102	2 Sep	38,073	947	7 Sep	31,260	2,334
29 Aug	17,523	3,798	3 Sep	16,011	3,702	8 Sep	15,261	4,311
30 Aug	12,827	2,862	4 Sep	14,375	2,461	9 Sep	20,801	3,140
31 Aug	14,792	771	5 Sep	26,504	2,533	10 Sep	28,232	6,949
1 Sep	11,938	3,610	6 Sep	30,538	1,818	11 Sep	57,679	5,817

Table 4. Requests per UTC day, all clients and crawler-token clients.

5. How much of the crawler traffic is genuine

Of the 46,155 crawler-token requests, 29,691 (64.3%) arrived through Cloudflare edge addresses and 16,464 (35.7%) reached the origin directly. A crawler that resolves the public hostname connects to Cloudflare (Section 2), so direct connections to the origin most likely come from clients that use a cached or scanned origin address. For the nine tokens whose operators publish address ranges, plus Google-Extended checked against Google's ranges, *none* of the direct requests came from an address inside those ranges (0 of 11,276). Direct requests never asked for `/robots.txt` in the post-deployment days: all 698 of those requests came through the CDN. A second signal points the same way. Google documents Google-Extended as a robots.txt product token that is not sent as a User-Agent [5], yet 863 requests presented it as one (none of them also contained Googlebot). Such requests are very likely imitations.

So a large share of the "crawler" traffic here is probably not what it claims to be. For the main result this matters less than it seems. The zero covers all 46,155 requests, so it covers any subset of them, genuine or not. What imitation does change is the comparison: the post-deployment CDN subset alone still made 698 requests for `/robots.txt` and 282 for `/sitemap.xml` out of 12,921, so the robots and sitemap activity does not rest on

the direct traffic. The per-token counts in Table 2 should be read as requests *presenting* a token, not as verified operator activity.

6. Correction to Internet-Draft -00

Section 2 of draft-arsentev-llm-context-discovery-00 reported the same measurement with smaller numbers. When I rebuilt the analysis from the preserved raw logs, those numbers did not reproduce. The draft's request total (331,758) is reached exactly when the preserved log is cut during the second 12:09:18 UTC on 11 September, which indicates that the draft was computed on the live log before the window closed, not on the complete copy taken on 12 September. This cut was found after the fact by searching for the draft's total; it was not recorded at the time. The cut does not explain everything. The draft's count of 56 context-file requests is not reproduced by any single cut (49 at 12:09; 56 is first reached at 12:21:53), so that figure came from a separate, unrecorded computation. The draft also names a list of 22 tokens that was not saved with the analysis, and differences in the crawler counts may come from that list. Table 5 uses a cut at 12:10:00 for the truncated variant. Three statements of the draft also need correcting. It describes the setting as "two hosts", but the shared log pooled three hostnames (Section 2). It says 40 of 56 context-file requests came from a command-line client "operated by the publisher" and the rest from ordinary browsers; the complete log gives 53 of 67 from command-line clients, 13 browser-like and 1 self-declared bot, and the attribution to the publisher cannot be checked (Section 4.2). And its reading that "the crawlers were present, were active" and "had no way to learn" of the file goes beyond the data: much of the crawler-token traffic is probably imitation (Section 5), and whether the files were linked was not recorded (Section 2). None of this changes the draft's central observation: in every variant, crawlers made zero requests for the context files.

Quantity	Draft -00	Log cut at 11 Sep 12:10 UTC	Complete window (this report)
Requests in window	331,758	331,808	345,808
Crawler-token requests, full window	44,005	44,126	46,155
Crawler-token requests, 9 to 11 Sep	13,917	13,877	15,906
/robots.txt, 9 to 11 Sep	577	572	698
Distinct tokens requesting /robots.txt	12	11	11
/sitemap.xml, 9 to 11 Sep	249	249	282
Context-file requests, all clients	56	49	67
Of those, from command-line clients	40	n/a	53
Crawler requests for context files	0	0	0

Table 5. Draft -00 figures against the two reproducible variants. The figures in the right-hand column replace those in the draft; a revised draft will cite this report.

7. Limitations

- **Short exposure.** The files were first observed being served about 48 hours (`llms.txt`) and 27 hours (`llms-full.txt`) before the window closed; the actual deployment times were not recorded. A crawler

that tries new paths on a weekly or monthly cycle would not show up. For the first 13 days of the window there was nothing at the path to retrieve.

- **One origin.** One server, one operator, modest traffic. Crawler behaviour varies with a site's traffic and authority, so this is one observation, not an estimate for the web.
- **Origin only.** Requests answered from the CDN cache are not in the log. Robots counts are lower bounds; the cache state of the context files during the window was not recorded.
- **Pooled hostnames.** The log does not record the hostname. Robots and sitemap counts include the second site, which has no context file. The zero for context files does not depend on this.
- **Unverified identity.** Attribution is by User-Agent. Section 5 shows that much of it is probably imitation. An AI system that fetched the file under a browser User-Agent would appear in Table 3 as browser-like and would not be identified.
- **Links not recorded.** I did not record whether any HTML page linked to the files during the window. The result therefore cannot separate a file nobody linked to from a link that crawlers did not follow. It says nothing about files that are linked prominently.
- **Token list.** The list is a choice. Operators that do not use one of the 22 tokens are not counted as crawlers. All 67 context-file requests are listed by class in the released data, so no retrieval by any client is hidden.
- **Ranges after the fact.** Operator ranges were fetched 17 days after the window. A range that was retired in that time would be missed. This would only matter if some direct requests were genuine, and it does not affect the CDN subset.

8. Data and code

The Zenodo record (DOI above) contains this report and the following files: `analyze.py`, `supplement.py`, `full-window.json`, `cutoff-1210.json`, `supplement.json`, `crawler_requests_by_day_token.csv` (day × token × requests), `per_token.csv`, `daily_totals.csv`, `context_file_requests.csv` (day × path × status × client class), `cut_search.py`, `cutoff-120918.json`, the snapshot of operator and Cloudflare address ranges used in Section 5 (`ranges-2026-09-28.zip`), and `SHA256SUMS` of the 15 raw log files. Raw logs are kept by the author and are not released because they contain visitor IP addresses. Research index: https://arsentev.ai/research.

References

1. J. Howard. The /llms.txt file. Proposal, 2024. https://llmstxt.org/
2. M. Koster, G. Illyes, H. Zeller, L. Sassman. Robots Exclusion Protocol. RFC 9309, September 2022.
3. Sitemaps XML format, protocol version 0.9. https://www.sitemaps.org/protocol.html
4. E. Arsentev. Discovery of LLM Context Files. Internet-Draft draft-arsentev-llm-context-discovery-00, IETF Datatracker, September 2026.
5. Google Search Central. Google's common crawlers and product tokens (Google-Extended); Googlebot and special-crawler IP range files. Accessed 28 September 2026.
6. OpenAI. Overview of OpenAI crawlers, and published IP range files gptbot.json, searchbot.json, chatgpt-user.json. Accessed 28 September 2026.

7. Cloudflare. IP ranges (ips-v4, ips-v6). Accessed 28 September 2026.
8. Microsoft Bing. bingbot.json IP range file. Accessed 28 September 2026.
9. Apple. About Applebot; applebot.json IP range file. Accessed 28 September 2026.
10. Perplexity. Perplexity crawlers; perplexitybot.json and perplexity-user.json IP range files. Accessed 28 September 2026.
11. DuckDuckGo. DuckDuckBot; duckduckbot.json IP range file. Accessed 28 September 2026.