TYGART MEDIA · NOTES FROM THE BOT LAB

The Bots Are Lying About Who They Are

By William Tygart · November 2026

One IP address came to my insurance site wearing four different name tags.

In 55 days of origin server logs from Risk Coverage Hub, the address 185.213.175.37 showed up identifying itself as GoogleOther. Then as bingbot. Then as PerplexityBot. Then as Applebot. Four trusted identities — crawlers that sites like mine deliberately make room for — all claimed by a single hostile scanner address, rotating costumes between requests.

None of those claims cost it anything. That's the point of this article: a user-agent string is not an identity. It's a line of text the visitor types about itself. Anyone can write "PerplexityBot" into a header in about four seconds, and a meaningful share of the traffic wearing the good bots' names is doing exactly that.

Why the costume works

Most sites meet crawlers at the door with a list. The robots file says who may crawl. The firewall and the CDN rules say whose requests get the fast lane, the higher rate limits, the exemption from bot challenges. And in a lot of stacks — more than anyone in this industry likes to admit — those decisions get made on the user-agent string, because it's the easiest field to match.

String matching is trust by assertion. If your allowlist says "let PerplexityBot through," what it actually says is "let through anyone willing to type PerplexityBot." The scanner at 185.213.175.37 understood that perfectly. It wasn't trying to defeat serious verification. It was betting there wasn't any — that the name tag alone would get it treated like a welcome guest while it did whatever it came to do.

This matters more every quarter, because the gap between the costumes is widening. A real search crawler is a guest you want: it indexes, it retrieves, it sends back buyers. A scanner wearing that crawler's name is probing for the soft parts of your stack. Treat them the same and you have not made a neutral error. You have handed your trust, wholesale, to whoever bothers to lie politely.

How you actually check

Real verification is unglamorous and it has three parts. None of them is the user-agent string.

One: the address. The major crawler operators publish the IP ranges their bots crawl from. A request claiming to be from one of those operators either originates inside the published ranges or it doesn't. There is no third state. 185.213.175.37 is not a Google address, a Microsoft address, an Anthropic address, or an Apple address, no matter what its header said that hour.

Two: the name resolves. Reverse DNS on the requesting IP should land on a domain the operator controls — the crawler's own hostname pattern — and a forward lookup on that hostname should return the same IP you started with. Operators document this because they *want* to be verified. A costume can't pass it; the scanner doesn't control the real operator's DNS.

Three: the behavior matches the job. A real indexer walks systematically and predictably. A retrieval fetcher arrives prompt-driven, aimed at specific pages. A scanner rotates identities, hammers the login door, and probes for config files. Even before you run the first two checks, the behavior of that address — four identities, one IP — had already confessed.

Do those three checks and the costume falls off. Skip them and you're not running bot management. You're running a guest list written by the guests.

The door they actually want

Here's why the lying isn't just an analytics problem. While costume traffic walks the public halls, a second, dumber, more patient assault runs against the one door that matters: the login page.

Across my own sites in the same 55-day window:

- **tygartmedia.com:** /wp-login.php took 48,164 hits, including 16,805 POSTs — those POSTs are not visits, they're login attempts, credentials being tried. Separately, /.env was probed 798 times by things hoping a configuration file full of secrets was sitting in the webroot.
- **Continuity Hub:** 11,798 hits on wp-login, 9,120 of them POSTs.
- **Risk Coverage Hub:** 4,759 login POSTs.
- **Healthcare Facilities Hub:** 7,307 wp-login hits, 4,359 POSTs.

Add it up and you're looking at tens of thousands of automated attempts in under two months, across four ordinary publications — not a bank, not a hospital system, just content sites. This is not a targeted campaign. It's weather. It runs all day, every day, against every WordPress login on the public internet, and the only question your logs answer is whether it's getting anywhere.

Mine say it isn't. Successful attacker logins on tygartmedia: zero. Secrets leaked from those 798 /.env probes: zero. Continuity Hub breaches: zero — the only successful logins in the entire window were 16 redirects from my own IP addresses. Risk Coverage's 10 successful logins: all operator IPs. Healthcare Facilities Hub: zero breaches. Four sites, 55 days, zero doors opened.

What zero actually means

I want to be precise about that zero, because it's easy to read wrong. Zero breaches is not evidence that the assault is harmless, and it's not evidence that my sites are hardened masterpieces. It's evidence that the boring work functions: rate limits on the login endpoint, current software, strong authentication, secrets kept out of the webroot. The unglamorous layer did its job 48,164 times without anyone watching.

It is also exactly why the costume problem deserves more respect than it gets. The defenses that held my login door are verification-based — they check things an attacker can't fake cheaply. The moment a stack substitutes name-tag trust anywhere else — an allowlist here, a rate-limit exemption there, a "verified bot" dashboard label taken at face value — it has built a second door and left it on the latch for anyone who can spell PerplexityBot.

Your logs already know which bots are lying to you. The IP is in there. The rotating headers are in there. The login attempts are in there, counted to the unit. You just have to read them as an identity problem instead of a traffic report.

We do AI citation restoration. When the AI can't find you, can't read you, or keeps citing somebody else — send us the worst one you have. Your RFPs, your disasters, the problem everyone else already failed to fix. We come from the restoration industry. We know what's on the floor, and cleaning it up is the job. And if you're a developer who just needs help on the hard part, get a hold of us.

— William Tygart, Tygart Media. Data: origin server logs, six-site lab, Sep 18–Nov 12, 2026.