

WUNDERLANDMEDIA

The Agent-Readiness Site Audit

An AI agent just filled out your contact form. Did anything break? Run this audit in an afternoon and find out.

By Kemal | wunderlandmedia.com

57.5%

Automated share of HTML traffic

1,700%

Agent request growth YoY

70%

AI traffic with no referrer

1 You Cannot Cleanly Detect This

57.5% Cloudflare Radar, which sees roughly a fifth of all websites, now puts automated requests at 57.5% of HTML traffic against 42.5% from humans. Agent requests specifically are up more than 1,700% year over year.

Before you plan anything, accept the constraint: you almost certainly cannot tell which visits to your site were agent-driven. Every fix in this audit is designed for the case where you never find out.

WHY IDENTIFICATION FAILS

- ☐ **Atlas and Comet are Chromium**
ChatGPT Atlas and Perplexity Comet fingerprint almost identically to a generic Chrome install under basic detection.
- ☐ **Gemini in Chrome and Copilot in Edge are worse**
They are not separate browsers at all. They are features inside a browser your visitors already use, with no distinct user agent string to filter on.
- ☐ **Analytics will say "Chrome" and be telling the truth**
There is no dimension in GA4 that reliably separates an agent session from a human one. Do not build a plan on a segment that does not exist.
- ☐ **Roughly 70% of AI-adjacent traffic arrives with no referrer**
This is why people call it dark traffic. Referrer-based attribution misses most of it by design.

HINTS YOU CAN LOOK FOR ANYWAY (HINTS, NOT PROOF)

- ☐ **Search the GA4 browser dimension for Atlas, Comet, PerplexityBot, CFNetwork**
Atlas sessions often carry CFNetwork and Darwin patterns server-side because of how OpenAI's networking layer wraps outbound requests.
- ☐ **Check referrers from perplexity.ai and the OpenAI domains**
A small, non-representative slice will still be attributed. Treat any number you get as a floor, never a measurement.
- ☐ **Note the stripped-Chromium tells**
Stripped builds miss extension manifest entries and Web Speech API voice lists that full Chrome exposes. Useful signal, unusable as a gate.

RIGHT APPROACH

Make the site work for everyone, agent or human, and stop trying to sort visitors into buckets you cannot verify.

WRONG APPROACH

"We will detect agents and serve them a special path." You will misclassify real customers and maintain two code paths.

Set the expectation now

The fix throughout this audit is never detection. It is removing the things that break for any visitor who is not holding a mouse. That happens to include agents, screen reader users, and anyone on a flaky connection.

"I have not measured this on a client site. I want to be straight about that. I do not have a 'we saw 4% of enquiries come from agents' number, and if someone hands you one, ask hard how they got it."

Kemal, Wunderlandmedia

2 Forms Audit

How an agent reads your form

Three channels: screenshots, raw HTML, and the accessibility tree. Anything that exists in only one of those, or only exists after a mouse does something, is a failure point.

MARKUP FUNDAMENTALS

- ☐ **Every input has a real <label for>**
Placeholder text is not a label. Without the association, the field has no accessible name and the agent is guessing from position.
- ☐ **The submit control is a real <button type="submit">**
A <div onclick> styled to look like a button is invisible as an action in the accessibility tree.
- ☐ **Input types are correct**
type="email", type="tel", type="url". The type tells the agent what shape of value belongs there.
- ☐ **Autocomplete attributes are present**
autocomplete="name", "email", "tel", "organization". Free semantics, and they help human autofill too.
- ☐ **Required fields are marked with the required attribute**
Not just a red asterisk in a span. The asterisk is a screenshot detail; the attribute is machine-readable.
- ☐ **Selects, radios and checkboxes use native elements**
Custom dropdowns built from divs need full ARIA to be operable. Native elements are correct by default.

WORKS

```
<label for="email">Email</label>
<input id="email" type="email" autocomplete="email" required>
<button type="submit">Send</button>
```

BREAKS

```
<input placeholder="Email">
<div class="btn" onclick="submitForm()">Send</div>
```

THINGS THAT ONLY EXIST FOR A MOUSE

- ☐ **Kill hover-revealed fields anywhere in a conversion path**
If a field only renders after a mouseover, it may as well not exist.
- ☐ **Kill scroll-revealed fields**
Fields injected when an intersection observer fires are missing from the first read of the page.
- ☐ **No elements a few pixels tall, and no transparent overlays over inputs**
Both make a field unclickable in practice while still appearing in the DOM.
- ☐ **No JavaScript-gated submit**
If the button only becomes real after a specific interaction sequence, the enquiry never leaves the page.

STRUCTURE AND FLOW

- ☐ **Collapse multi-step wizards to one step where you can**
Every step is another place to lose state, another dynamically injected field, another validation rule firing on blur. Simple forms survive; five-step qualification funnels with conditional logic do not.
- ☐ **Rename or remove honeypot fields**
A hidden input named "website" or "phone2" catches dumb spam bots beautifully. It also catches a language model that reads the DOM, sees a field called "website", and helpfully fills in the company URL because it is being thorough.
- ☐ **Prefer server-side scoring over a single hidden trap**
Field-level traps assume the filler is stupid. This one is not.
- ☐ **Errors are readable, specific and attached to the field**
A red border alone says nothing. Use aria-describedby and text that names what is wrong and how to fix it, so the agent can correct and retry instead of stalling.
- ☐ **Success is a visible, textual confirmation**
If the only feedback is a toast that vanishes in two seconds, nobody, human or agent, knows whether the enquiry sent.
- ☐ **Check the consent wall in front of the form**
An agent will click something. It will not necessarily click what the user would have clicked. If your cookie banner gates the form behind a choice, you now hold a consent record nobody meaningfully gave.

3

Bot Defence Audit

Your CAPTCHA is not asking "are you human"

Turnstile scores browser behaviour, device integrity and interaction patterns in the background, before any visible challenge. It is asking "does this session look normal," and a user-driven agent does not. A real customer with a real budget gets a hard stop and no explanation.

WHAT HAPPENS TO AN AGENT SESSION

- ☐ **The environment is flagged before the widget is drawn**
Reports from people running agent mode describe the challenge failing even when a human takes over and clicks it manually, because the browser environment was already flagged. Handing control back does not rescue the session.
- ☐ **Invisible modes fail invisibly**
With no interactive widget there is nothing to click and nothing to explain the block. The enquiry simply does not arrive and nobody tells you.
- ☐ **Image challenges are the wrong test for the threat you have**
LLM-written spam now arrives through the visible form, correctly filled, grammatical and personalised. The challenge is not what stops it.

Defence	How it treats an agent session	Verdict on a contact form
Cloudflare Turnstile	Scores the browser environment before any visible challenge; flagged sessions fail even with a human clicking	High risk of silent loss
hCaptcha	Behavioural and device signals plus visual challenge; automated interaction patterns score badly	High risk
reCAPTCHA v3	Returns a low score with no challenge to pass; your threshold silently decides	Review your threshold
reCAPTCHA v2 checkbox	Risk analysis first, then an image challenge if the session looks automated	Passable, often not
Server-side content scoring	Judges the message, not the mechanics of who typed it	Preferred

CHECK YOUR OWN CONFIGURATION

- ☐ **Find out which challenge is actually on the form**
Plugins and form builders often add one you did not choose. Look at the rendered page, not the settings screen.
- ☐ **Read your reCAPTCHA v3 score threshold**
If you are rejecting below 0.5 on a low-value contact form, you are throwing away enquiries to save yourself reading email.
- ☐ **Log rejected submissions instead of discarding them**
A quarantine folder you skim once a week beats a silent bin. This is the only way you ever learn what the challenge cost you.
- ☐ **Reconsider the CAPTCHA entirely**
If it is guarding a low-value contact form rather than a login or a payment, the enquiries it costs may already outweigh the spam it stops.
- ☐ **Filter on content and intent, not mechanics**
Does the message reference something specific a scraper would not know? Does the phone number work? Does the domain resolve to a real company?

The cheapest test there is

Does a reply get answered by a human within a day? That works regardless of what filled the form. Set the bar at "reply to everything that is not obviously junk." Losing one real customer costs more than reading fifty bad emails.

4

The Cloudflare Default Nobody Noticed

September 15, 2026: new crawler defaults

Training and Agent traffic is now blocked by default on pages that display ads, while Search stays allowed. This applies to new domains, new customers, and every existing customer on the free tier. Paid existing customers keep their old settings unless they opt in. Read that again if you are on the free tier: the Agent category is blocked by default, and that is the category that includes something acting in real time on a person's behalf.

KNOW THE THREE CATEGORIES BEFORE YOU TOUCH A SETTING

Category	What it means	Default after Sept 15
Search	Collecting and indexing content to answer questions later	Allowed
Training	Taking content to train a model	Blocked on ad-displaying pages
Agent	Automated behaviour acting in real time on a person's behalf	Blocked on ad-displaying pages

HOW TO CHECK WHETHER IT APPLIES TO YOU

- ☐ **Confirm which plan the zone is on**
Free tier means the new defaults were applied to you whether or not you read the announcement. Paid and pre-existing means you kept your old settings.
- ☐ **Check when the domain was added**
New domains and new customers get the new defaults regardless of plan.
- ☐ **Open the bot management and crawler controls and read the per-category state**
You are looking specifically at whether Agent is set to block. Do not assume from the plan alone.
- ☐ **Check whether your pages display ads**
The default is scoped to ad-displaying pages. A fifteen-page service site with no ads is a different calculation from a publisher.
- ☐ **Decide deliberately and write down why**
The defaults were chosen for ad-funded publishers. If you are trying to get enquiries rather than sell content, inheriting them is a choice you did not make.
- ☐ **Re-check after any plan change or zone migration**
Defaults reapply at the boundaries. A move to a new account can quietly reset what you configured.

PUBLISHER COVERED IN ADS

Blocking Training and Agent is probably right. Cloudflare rebranded Pay Per Crawl to Pay Per Use, with Ceramic.ai paying publishers when content appears in results and You.com paying when an agent accesses premium content. There is a real business reason to hold the line.

SERVICE BUSINESS, NO ADS

You are not selling content, you are trying to get enquiries. A blocked Agent category means a person who asked their browser to request a quote never reaches your form, and you never hear about it.

The tension is real

You want the scraper gone and the shopper through. Those are currently hard to separate. Anyone who tells you it is a solved problem is selling something.

5 Rate Limiting and Timeouts

2 sec

Time traps that reject anything completed in under two seconds have the same problem as aggressive rate limits. They were a genuinely good anti-spam technique until this year. A two-second minimum submission time is now a customer filter, not a spam filter.

Agents move fast and do not pause. Every defence you built around human tempo now fires on legitimate traffic, and it fires silently.

WHAT TRIPS

- ☐ **Per-IP submission windows**
A form allowing one submission per IP per thirty seconds is fine for humans and reads as an attack from something that filled six fields in two seconds.
- ☐ **Minimum time-on-form checks**
The timestamp gap between page load and submit is the classic trap. An agent closes it in a fraction of the threshold.
- ☐ **Burst limits at the edge**
A session that loads the page, the privacy policy and the pricing page in under a second can breach a WAF rate rule before it ever reaches the form.
- ☐ **CSRF and session token expiry**
Rare in the fast direction, but tokens tied to a narrow validity window can also break the other way when an agent pauses mid-task and resumes later.
- ☐ **Debounced or throttled field validation**
Validation that only fires 500ms after typing stops may never run if fields are set programmatically, leaving the form in a permanently invalid state.
- ☐ **Shared-IP aggregation**
Agent traffic can originate from provider infrastructure, so several unrelated users land on the same address and trip a per-IP counter together.

WHAT TO CHECK ON THE FORM ENDPOINT

- ☐ **Find the actual limit, not the one in the docs**
Check the host, the WAF, the framework middleware and the form plugin. Most sites have three layers of limiting and the owner knows about one.
- ☐ **Make sure a fast legitimate submission is not treated as abuse**
Submit the form yourself as quickly as you physically can and confirm it goes through.
- ☐ **Loosen or remove the time trap**
If you keep it, drop the threshold to something a fast human could not beat and an agent will not either, and log rather than reject.
- ☐ **Return a readable error, never a silent drop**
A 429 with a human-readable message lets the sender retry. A blank page or a fake success screen loses the enquiry permanently.
- ☐ **Alert on rejection volume**
If your form endpoint starts rejecting more than it accepts, you want to know that week, not next quarter.

DO

Log the fast submission, score it on content, and let it through. Rate limit the endpoint generously and alert on genuine floods.

DO NOT

Reject anything submitted in under two seconds and show a generic "something went wrong." That is a hard stop for a real buyer, with no explanation.

6

The Accessibility Overlap

The insight that makes this tractable

Agents read the accessibility tree. That is the same browser-native layer of roles, names and states that assistive technology has used for twenty years. OpenAI's publisher documentation says Atlas uses ARIA tags, the same labels and roles that support screen readers. Google's guidance for agent-friendly sites says the same thing in different words. Read that list and it is a WCAG audit. Every single item.

THE COMBINED CHECKLIST: FIX ONCE, PAY OFF TWICE

Fix	What it does for a screen reader	What it does for an agent
Real <label for> on inputs	Announces what the field is	Gives the field an accessible name to target
Real <button> and <a>	Announces role and is keyboard operable	Exposes the action as an action
No hover-only interactions	Content reachable without a pointer	Content exists on first read of the page
Stable layouts	Focus does not jump mid-task	Element positions stay valid between steps
Visible interactive elements	Nothing hidden behind an overlay	Nothing unclickable that appears clickable
Clear focus states	Keyboard users know where they are	Confirms which control is active
Descriptive error text	Says what is wrong and how to fix it	Lets the sender correct and retry

RUN IT AS ONE JOB

- ☐ **Run an accessibility audit against WCAG 2.2 and treat the output as your agent compatibility report**
It very nearly is. Start with the conversion paths: home, services, contact.
- ☐ **Fix missing labels first**
Highest ratio of effort to breakage removed, on both sides.
- ☐ **Replace unlabelled icon buttons with named ones**
An icon-only close or submit button with no accessible name is an unusable control in both worlds.
- ☐ **Untangle div-soup navigation**
Real nav, real lists, real links. Breaks a screen reader and an agent in the same way for the same reason.
- ☐ **Fix unclear focus states**
Never remove the focus ring without replacing it with something at least as visible.
- ☐ **Tab through the whole form with the mouse untouched**
If you cannot complete it that way, neither can an agent, and neither can a chunk of your human audience.

"You do not need a new discipline, a new vendor, or an 'agent readiness' consultant. You need the accessibility work that has been on your backlog since 2023. That is a rare thing in this industry: two problems, one fix, and the fix was already worth doing on its own merits."

Kemal, Wunderlandmedia

7 Signed Agents and What Is Coming

Watch this, do not act on it yet

Nothing in this section is work for this afternoon. It is the shape the solution is taking, so you recognise it when it lands and do not buy something premature in the meantime.

WEB BOT AUTH

- ☐ **An IETF draft applying HTTP Message Signatures to automated traffic**
Built on RFC 9421, so an agent can cryptographically prove who it is instead of asserting it in a header that anyone can copy.
- ☐ **This is the right shape of solution**
It replaces "trust this user agent string" with something verifiable, which is the only way the scraper-versus-shopper problem ever gets separated cleanly.
- ☐ **It is also early, partial, and covers a handful of agents**
The bulk of AI agents using websites in the wild right now are not signed and will not be for a while.

THE SIGNED-AGENT COHORT

Programme member	What it is
ChatGPT agent	OpenAI's agent mode
Goose (Block)	Open-source agent framework
Browserbase	Hosted browser infrastructure for agents
Anchor Browser	Agent-oriented browser platform
Cloudflare Browser Rendering	Cloudflare's own headless rendering service

THE PAYMENT LAYER

- ☐ **Visa folded the Trusted Agent Protocol into Visa Intelligent Commerce**
Agent identity as a first-class part of the payment flow.
- ☐ **Mastercard built the same idea into Agent Pay**
Same premise, separate implementation.
- ☐ **American Express committed to it for its agentic commerce programme**
All of it sits on top of Cloudflare's Web Bot Auth work.

3 of 3 Visa, Mastercard and Amex have all committed. Nobody builds a payment authentication layer for traffic that has no money behind it. That is the strongest available signal that this category matters, and it is a signal about intent, not a conversion rate.

WHAT NOT TO DO WHILE YOU WAIT

- ☐ **Do not block agents wholesale**
The mistake I expect to see most this year, and it is expensive in a way that never shows up in a report, because nobody sends you an email to say they could not send you an email.
- ☐ **Do not build agent-only shortcuts or hidden endpoints**
You will maintain two paths and one of them will silently rot.
- ☐ **Do not buy "AI agent optimisation" before running an accessibility audit**
The audit is cheaper and it is most of the answer.
- ☐ **Do not confuse this with letting your team run agents**
Letting agents use your site is a different question from letting staff run agents inside your browser sessions. That second one is a genuine security problem.

The Afternoon Audit

Check	How	Pass criteria
Live agent run	Open your main contact form in ChatGPT Atlas or Perplexity Comet with agent mode on and tell it to submit an enquiry	The enquiry lands in your inbox
Cloudflare bot settings	Check plan, domain age and the per-category state in crawler controls	Agent category state is a decision you made
Form markup	Inspect every input: label for, button type, input type, autocomplete	No <div onclick>, no placeholder-as-label
Hover and scroll reveals	Load the conversion path with JavaScript throttled and no pointer movement	Every field present on first render
Honeypot fields	Read the field names in the DOM as if you were being helpful	No trap named like a real field
Time trap	Submit as fast as you physically can	Fast submission accepted
Rate limits	Check host, WAF, framework middleware and form plugin separately	Legitimate speed is not abuse
CAPTCHA	Identify which one is on the form and read the score threshold	Justified by the value it guards
Multi-step forms	Count the steps and the conditional rules	One step wherever possible
Accessibility audit	Run against WCAG 2.2 on home, services and contact	Labels, roles, focus states clean
Keyboard-only pass	Complete the form without touching the mouse	Submission completes
Reply policy	Reply to everything that is not obviously junk, within a day	A human answers

"I do not know how much of this matters yet in raw numbers, and I would rather say that than make one up. What I do know is that the cost of being ready is low, the work is already justified for human visitors, and the downside of not being ready is a customer who quietly went somewhere else."

Kemal, Wunderlandmedia

If you only do one thing

Run the accessibility audit and fix what it finds. That is most of the work, it was already on your backlog, and it fixes the agent problem as a side effect. WebMCP is a project. Fixing your form labels is a Tuesday.

Honest limits of this document

Nobody has reliable conversion data on agent-driven enquiries yet. The behaviour is roughly a year old at scale, there is no reliable identification, so there is no reliable cohort and no reliable conversion rate. Anyone selling you one built it on guesswork. I have not measured agent traffic on a client site either.

Read the Full Post

wunderlandmedia.com/ai-agents-using-websites-contact-forms

Want someone to run this checklist against your site and tell you what is actually broken? That is the kind of work I do. One person, no upsell, honest answer either way. wunderlandmedia.com